

**APLIKASI PEMESANAN TRAVEL MENGGUNAKAN
CHATBOT DENGAN MACHINE LEARNING**

SKRIPSI



Oleh :

**ADHINDYA TRIGHA NUGRAHA
NIM. 15650125**

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2020**

**APLIKASI PEMESANAN TRAVEL MENGGUNAKAN CHATBOT
DENGAN MACHINE LEARNING**

SKRIPSI

**Diajukan kepada:
Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang
Untuk Memenuhi Salah Satu Persyaratan Dalam
Memperoleh Gelar Sarjana Komputer (S.Kom)**

**Oleh:
ADHINDYA TRIGHA NUGRAHA
NIM. 15650125**

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2020**

LEMBAR PERSETUJUAN
APLIKASI PEMESANAN TRAVEL MENGGUNAKAN
CHATBOT DENGAN MACHINE LEARNING

SKRIPSI

Oleh :
ADHINDYA TRIGHA NUGRAHA
NIM. 15650125

Telah Diperiksa dan Disetujui untuk Diuji
Tanggal : 22 Mei 2020

Dosen Pembimbing I

Dosen Pembimbing II

A'la Syauqi, M.Kom
NIP. 19771201 200801 1 007

Yunifa Miftachul Arif, M.T
NIP. 19740602 200901 1 010

Mengetahui,
Ketua Jurusan Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang

Dr. Cahyo Crysdian
NIP. 19740424 200901 1 008

LEMBAR PENGESAHAN

APLIKASI PEMESANAN TRAVEL MENGGUNAKAN CHATBOT DENGAN MACHINE LEARNING

SKRIPSI

Oleh:
ADHINDYA TRIGHA NUGRAHA
NIM. 15650125

Telah Dipertahankan di Depan Dewan Penguji
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
untuk Memperoleh Gelar Sarjana Komputer (S.Kom)
Pada Tanggal Juli 2020

Susunan Dewan Penguji		Tanda tangan
1. Penguji Utama	<u>Dr. M. Amin Hariyadi, M.T</u> : NIP. 19670118 200501 1 001	()
2. Ketua Penguji	<u>Fajar Rohman Hariri, M. Kom</u> : NIP. 19890515 201801 1 001	()
3. Sekretaris Penguji	<u>A'la Syauqi, M. Kom</u> : NIP. 19771201 200801 1 007	()
4. Anggota Penguji	<u>Yunifa Miftachul Arif, M.T</u> : NIP. 19830616 201101 1 004	()

Mengetahui,
Ketua Jurusan Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang

Dr. Cahyo Crysdian
NIP. 19740424 200901 1 008

PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan dibawah ini:

Nama : Adhinya Trigha Nugraha

NIM : 15650125

Fakultas/Jurusan : Sains dan Teknologi/Teknik Infomatika

Judul Skripsi : Aplikasi Pemesanan Travel Menggunakan Chatbot Dengan
Machine Learning

Menyatakan dengan sebenarnya bahwa Skripsi yang saya tulis ini benar-benar merupakan hasil karya sendiri, bukan merupakan pengambilalihan data, tulisan atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka.

Apabila dikemudian hari terbukti atau dapat dibuktikan Skripsi ini hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 22 Juni 2020
Yang membuat pernyataan,



Adhinya Trigha Nugraha
NIM. 15650125

HALAMAN MOTTO



HALAMAN PERSEMBAHAN

الْحَمْدُ لِلَّهِ رَبِّ الْعَالَمِينَ

Atas kehadiran Allah Subhanahu wa ta'ala, dengan mengucapkan syukur alhamdulillah penulis mempersembahkan sebuah karya untuk orang – orang yang sangat berarti

Terima kasih saya ucapkan kepada kedua orang tua yang telah dan selalu memberikan dukungan, motivasi dan doa kepada saya dan selalu mengajarkan serta mendidik berbagai nilai – nilai dalam kehidupan.

Terima kasih juga diucapkan kepada kedua ibu pembimbing yang telah membimbing dalam melakukan penelitian dan memberikan motivasi serta dorongan hingga penelitian terselesaikan dengan lancar.

Tak lupa juga diucapkan terima kasih kepada sahabat – sahabat saya yang telah mendukung serta mendoakan selama saya masuk perkuliahan di UIN Maulana Malik Ibrahim Malang. Dan juga keluarga besar Teknik Informatika Interface 2015 yang telah memberikan semangat dan doa – doa yang dikirimkan.

Terima kasih untuk orang – orang yang tidak bisa disebutkan satu per satu yang telah mendoakan serta mendukung sehingga dapat terselesaikan skripsi ini.

KATA PENGANTAR

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Puji dan syukur saya panjatkan ke hadirat Allah subhanahu wa ta'ala yang telah melimpahkan rahmat dan hidayahNya kepada kita, sehingga saya dapat menyelesaikan skripsi dengan tepat waktu, yang saya beri judul “Simulator NPC Berbasis Fuzzy *Sugeno* Untuk Pembelajaran Edukasi Mitigasi Pasca Bencana Tsunami”. Tujuan dari penyusunan skripsi ini guna memenuhi salah satu syarat untuk bisa menempuh ujian sarjana komputer pada Fakultas Sains dan Teknologi (FSAINTEK) Program Studi Teknik Informatika di Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang. Di dalam pengerjaan skripsi ini telah melibatkan banyak pihak yang sangat membantu dalam banyak hal. Oleh sebab itu, disini penulis sampaikan rasa terima kasih sedalam-dalamnya kepada:

1. Prof. Dr. Abdul Haris, M.Ag selaku Rektor Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang.
2. Dr. Sri Harini, M.Si, selaku Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang.
3. Dr. Cahyo Crysdiyan, Selaku Ketua Jurusan Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang.
4. A'la Syauqi, M.Kom, selaku Dosen Pembimbing I yang telah membimbing dalam penyusunan skripsi ini hingga selesai.
5. Yunifa Miftachul Arif, M.T, selaku Dosen Pembimbing II yang telah membimbing dalam penyusunan skripsi ini hingga selesai.

6. Ainatul Mardhiyah, S.Kom, M.Cs, selaku Dosen Wali yang senantiasa memberikan banyak motivasi dan saran untuk kebaikan penulis.
7. Kedua orang tua tercinta Bapak Sumaryono dan Ibu Nur Asmi yang telah banyak memberikan doa dan dukungan kepada penulis secara moril maupun materil hingga skripsi ini dapat terselesaikan.
8. Anggota keluarga dan kerabat yang senantiasa memberikan doa dan dukungan semangat kepada penulis.
9. Sahabat – sahabat seperjuangan yaitu Fio Aziz, Faizuddin Pulungan, Afifudin Zuhri , Barly Vallendito, Julian Yusuf, Zaki Rozini, Aghna, Hazmi, Bayu, Aufa, Laode, dan Izzil yang tiada henti memberi dukungan dan motivasi kepada penulis serta target bersama untuk lulus skripsi dan wisuda bersama.
10. Sahabat – sahabat Interface Teknik Informatika 2015 yang selalu memberikan semangat dan doa kepada penulis.
11. Semua pihak yang telah banyak membantu dalam penyusunan skripsi ini yang tidak bisa penulis sebutkan semuanya.

Penulis menyadari bahwa dalam penyusunan skripsi ini masih terdapat kekurangan dan penulis berharap semoga skripsi ini bisa memberikan manfaat kepada para pembaca khususnya bagi penulis secara pribadi.

Malang, Juli 2020

Penulis

DAFTAR ISI

HALAMAN PENGAJUAN.....	i
LEMBAR PERSETUJUAN.	ii
LEMBAR PENGESAHAN.	iii
PERNYATAAN KEASLIAN TULISAN.	iv
HALAMAN MOTTO.....	iv
HALAMAN PERSEMBAHAN.	vi
KATA PENGANTAR	vii
DAFTAR ISI.....	ix
DAFTAR GAMBAR.....	xi
DAFTAR TABEL.....	xiii
ABSTRAK.....	xiv
ABSTRACT.....	xv
الملخص	xvi
BAB I.....	1
PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Pernyataan Masalah.....	3
1.3. Tujuan Penelitian.....	3
1.4. Batasan Masalah.....	4
1.5. Manfaat Penelitian.....	5
1.6. Sistematika Penulisan.....	5
BAB II.....	7
TINJAUAN PUSTAKA	7
2.1. Penelitian Terkait	7
2.2. Landasan Teori	9

2.2.1.	Chatbot	9
2.2.2.	Preprocessing	11
2.2.3.	Neural Network	13
BAB III	19
METODOLOGI PENELITIAN	19
3.1.	Objek Penelitian	19
3.2.	Desain Aplikasi	19
3.3.	Desain Proses	20
3.3.1.	Input Kalimat	20
3.3.2.	Preprocessing	21
3.3.3.	Training Data	26
3.3.4.	Output	36
3.4.	Implementasi Desain Antarmuka	37
3.4.1.	Halaman Beranda	37
3.4.2.	Chatbot Messenger	38
BAB IV	42
UJI COBA DAN PEMBAHASAN	42
4.1.	Skenario Pengujian	42
4.1.1.	Spesifikasi Hardware	42
4.1.2.	Spesifikasi Software	42
4.2.	Pengujian Sistem	43
BAB V	48
KESIMPULAN DAN SARA	48
5.1.	Kesimpulan.....	48
5.2.	Saran.....	48
DAFTAR PUSTAKA	499

DAFTAR GAMBAR

Gambar 2.1 Contoh <i>Case Folding</i>	11
Gambar 2.2 Contoh <i>Tokenizing</i>	12
Gambar 2.3 Contoh <i>Stemming</i>	12
Gambar 2.4 Ilustrasi Kemiripan Neural Network dan Neuron Otak.	13
Gambar 2.5 <i>Neural Network</i> 2 Hidden Layer.....	14
Gambar 2.6 Model Sederhana <i>Forward Pass</i>	15
Gambar 2.7 Proses Training Input ke Layer Pertama.....	16
Gambar 2.8 Proses Training Layer Pertama ke Layer Kedua.....	17
Gambar 2.9 Proses Training Layer Kedua ke Output.....	18
Gambar 3.1 Alur Desain Proses.....	20
Gambar 3.2 Alur <i>Preprocessing</i>	21
Gambar 3.3 Proses <i>Case Folding</i>	21
Gambar 3.4 Hasil <i>Case Folding</i>	22
Gambar 3.5 Proses <i>Tokenizing</i>	23
Gambar 3.6 Hasil <i>Tokenizing</i>	24
Gambar 3.7 Proses <i>Stemming</i>	25
Gambar 3.8 Hasil <i>Stemming</i>	26
Gambar 3.9 Proses Perubahan Angka.....	27
Gambar 3.10 Contoh Perubahan Angka	27
Gambar 3.11 Data dalam Bentuk File Json.....	28
Gambar 3.12 Training Input ke Layer Pertama	29
Gambar 3.13 Training Layer Pertama ke Layer Kedua	30
Gambar 3.14 Training Layer Kedua ke Output	31
Gambar 3.15 Model Training Neural Network.....	32
Gambar 3.16 Perubahan Nilai	33
Gambar 3.17 Proses Pelatihan	34
Gambar 3.18 Penyimpanan Model Training	34
Gambar 3.19 Grafik Akurasi Pelatihan.....	35
Gambar 3.20 Kode Program Ranking Jawaban	36
Gambar 3.21 Perbedaan Penggunaan Kata	37
Gambar 3.22 Halaman Utama.....	38
Gambar 3.23 Potongan Kode Program List Pertanyaan	39

Gambar 3.24 Potongan Kode Program Pencocokan Pertanyaan	40
Gambar 3.25 Database Penyimpanan Sementara	40
Gambar 3.26 Tampilan Chatbot.....	41
Gambar 4.1 Akurasi Pengujian yang Digunakan.....	45
Gambar 4.2 Perhitungan Klasifikasi	46
Gambar 4.3 Ranging Klasifikasi.....	46
Gambar 4.4 Proses Mendapatkan Jawaban.....	47



DAFTAR TABEL

Tabel 3.1 Nilai Parameter Proses Pelatihan	34
Tabel 3.2 Akurasi yang Didapatkan.....	34
Tabel 4.1 Pertanyaan yang Sering Diajukan User	44
Tabel 4.2 Penilaian Akurasi	46



ABSTRAK

Adhindya Trigha Nugraha. 2020. *Aplikasi Pemesanan Travel Menggunakan Chatbot Dengan Machine Learning*. Skripsi. Jurusan Teknik Informatika Fakultas Sains Dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing : (I) A'la Syauqi, M.Kom (II) Yunifa Miftachul Arif, M.T.

Kata Kunci : *neural network, chatbot, pemesanan travel.*

Chatbot merupakan layanan obrolan yang sudah sering dijumpai di berbagai macam platform, sesuai dengan namanya yang membalas dalam proses *chatting* adalah robot atau tokoh virtual. Chatbot diberikan kecerdasan buatan atau AI (*Artificial Intelligence*) yang mampu menirukan percakapan manusia seperti bercakap dengan manusia biasa. Kemampuan AI yang mampu memberikan kecerdasan virtual kepada *chatbot* yang dapat membuatnya menilai jawaban yang paling tepat untuk pertanyaan yang diberikan, dengan adanya *keyword chatbot* dapat menentukan jawaban yang terbaik sehingga pertanyaan yang diberikan dapat dijawab dengan cepat. Ada beberapa teknologi yang digunakan *chatbot*, untuk menghasilkan informasi yang tepat digunakan metode *neural network* yang terdiri dari *input, preprocessing, dan output*. *Input* merupakan data dasar pemesanan travel yang digunakan untuk melatih *chatbot* sehingga dapat memberikan jawaban ketika diberi pertanyaan, untuk *input* lainnya merupakan data pertanyaan yang diajukan *user*. *Preprocessing* merupakan proses yang didalamnya untuk mengolah kata dengan menggunakan *case folding, tokenizing, dan stemming*, yang nantinya akan dilakukan pelatihan lebih lanjut menggunakan *neural network* dimana *chatbot* dilatih untuk dapat menjawab setiap pertanyaan yang diberikan. Setelah mengimplementasikan metode *neural network* ke dalam *chatbot*, hasil *output* yang diberikan sesuai dengan yang diinginkan. Untuk validasi metode yang diimplementasikan ke dalam *chatbot*, dilakukan uji coba metode *neural network* menggunakan Python. Python digunakan untuk melakukan 27 percobaan dan nilai keluaran yang dihasilkan pun sesuai. Perbedaan nilai persentase akurasi tidak cukup jauh yang mana *chatbot* tidak akan memberikan jawaban yang menyimpang. Nilai persentase akurasi yang didapatkan sebesar 83% ketika diajukan ke *user* dan 85% untuk pelatihan *chatbot*.

ABSTRACT

Nugraha, Adhinda Trigha. 2020. *Application for Booking Travel Using Chatbot with Machine Learning*. Essay. Department of Informatics Engineering, Faculty of Science and Technology, Islamic State University of Maulana Malik Ibrahim of Malang. Counselor: (I) A'la Syauqi, M.Kom (II) Yunifa Miftachul Arif, M.T.

Keywords : *neural network, chatbot, travel booking.*

Chatbot is a chat service that is often found on various platform, as the name implies behind the reply in the chat process is a bot or virtual figure. *Chatbot* is given Artificial Intelligence that can mimic human conversation like conversing with ordinary people. The ability of Artificial Intelligence that gives virtual intelligence to chatbot that can make choices the most correct answer, by using keyword chatbot can determine the best answer and can answered quickly. There are several technologies that chatbot use, like neural network in order to precise information there are input, preprocessing, and output. Input is the basic data of travel booking that is used to train chatbot to answer question, for other input is the question data asked by user. Preprocessing is a process for processing words by using case folding, tokenizing, and stemming which will be further training using neural network where chatbot is trained to answer any question from user that relate with traveling. After implementing neural network into chatbot, the output result that given are good. For the validation of method neural network into chatbot was tested using python, by using python to do 27 experiments and the output value is good. The difference percentage value of accuracy is not far enough that the chatbot will not give strange answers. The percentage of accuracy obtained is 83% when submitted to users and 85% for chatbot training.

المخلص

نوغرها, اندنبا تريغا. 2020. تطبيق حجز السفر باستخدام شات بوت مع التعلم الآلي. البحث العلمي. قسم تقنيّة المعلومات، كلية العلوم والتكنولوجيا، جامعة مولانا مالك إبراهيم الإسلامية الحكومية بمالانج.

المشرفة : (I) أعلى شوقي الماجستير, (II) يونيفا مفتاح العارف الماجستير

الكلمات الرئيسية : الشبكة العصبية, شات بوت, حجوزات السفر.

شات بوت هي خدمة دردشة تتم مصادفتها غالبًا على أنظمة أساسية مختلفة ، حيث يشير الاسم إلى أن الرد في عملية الدردشة هو روبوت أو شخصية افتراضية. يتم منح شات بوت ذكاء اصطناعي أو AI (ذكاء اصطناعي) قادر على محاكاة المحادثة البشرية مثل التحدث مع الناس العاديين. إن قدرة الذكاء الاصطناعي قادرة على توفير ذكاء افتراضي لبرامج الدردشة الآلية التي يمكنها أن تجعلها تحكم على الإجابات الأكثر ملاءمة للأسئلة المقدمة ، مع تحديد الكلمة الرئيسية لبرنامج شات بوت أفضل إجابة ، بحيث يمكن الإجابة على الأسئلة المقدمة بسرعة. هناك العديد من التقنيات التي تستخدمها برامج الدردشة الآلية لإنتاج المعلومات الصحيحة ، وإنتاج المعلومات الصحيحة ، تتكون طريقة الشبكة العصبية من الإدخال والمعالجة المسبقة والمخرجات. المدخلات هي البيانات الأساسية لحجز السفر والتي يتم استخدامها لتدريب برنامج الدردشة الآلي بحيث يمكنها تقديم إجابات عند طرح سؤال ، أما بالنسبة للمدخلات الأخرى ، فإن البيانات هي السؤال الذي يطرحه المستخدم. المعالجة المسبقة هي عملية يتم من خلالها معالجة الكلمات باستخدام طي الحالة ، والتشفير ، والجذع ، والتي سيتم إجراؤها لاحقًا لمزيد من التدريب باستخدام شبكة عصبية حيث يتم تدريب روبوتات الدردشة على الإجابة عن أي أسئلة مقدمة. بعد تنفيذ طريقة الشبكة العصبية في برنامج الدردشة الآلي ، يتم إعطاء نتائج الإخراج حسب الرغبة. للتحقق من صحة الأساليب المطبقة في شات بوت ، تم اختبار طريقة الشبكة العصبية باستخدام بايثون. يتم استخدام Python لإجراء 27 تجربة وقيمة المخرجات الناتجة مناسبة أيضًا. الفرق في قيمة النسبة المئوية للدقة ليس بعيدًا بما يكفي بحيث لا يعطي برنامج الدردشة إجابة مشوهة. النسبة المئوية للدقة التي تم الحصول عليها هي 83% عند تقديمها للمستخدمين و 85% لتدريب شات بوت.

BAB I

PENDAHULUAN

1.1. Latar Belakang

Perkembangan zaman membuat kemajuan teknologi berkembang secara pesat, salah satu yang saat ini sedang banyak diteliti yaitu kecerdasan buatan. Menurut John McCharty, 1956, AI : untuk mengetahui dan memodelkan proses – proses berfikir manusia dan mendesain mesin agar dapat menirukan perilaku manusia. Cerdas, berarti memiliki pengetahuan ditambah pengalaman, penalaran (bagaimana membuat keputusan dan mengambil tindakan), moral yang baik. Istilah kecerdasan buatan dapat digunakan untuk mengacu ke perangkat lunak yang memiliki kecerdasan, salah satu contoh program otonom yang digunakan melalui bantuan operator atau data mining disebut sebagai *bot*.

Chatbot merupakan sebuah program komputer yang dapat menjalankan *intelligent conversation* dengan pengguna melalui media suara maupun teks, sering kali dilakukan dengan melakukan percakapan singkat (Tillotson, 2012). *Chatbot* bekerja dengan mengartikan pesan yang diberikan oleh pengguna, kemudian memproses perkataan dari pengguna tersebut, selanjutnya menentukan dan mengeksekusi apa yang perlu *chatbot* lakukan berdasarkan perintah dari pengguna, dan terakhir menyampaikan hasil dari kesimpulan kepada pengguna (Shawar dan Atwell, 2007). Jadi jika pengguna sedang melakukan percakapan dengan program ini, maka pengguna seperti ada dua pribadi manusia yang saling berkomunikasi. Percakapan yang dilakukan *bot* tersebut tidak nyata adanya, karena *bot* tersebut sudah dirancang untuk merespon segala jenis pertanyaan dan pertanyaan diinputkan oleh manusia(pengguna). Hal ini terjadi karena sebelumnya

sudah dideklarasikan pada database, berupa entitas – entitas kata, pola kalimat, dan berbagai jenis pertanyaan dan pertanyaan.

Respon yang dihasilkan merupakan hasil pemindaian dari kata kunci pada masukan pengguna dan menghasilkan respon balasan yang dianggap paling cocok, atau pola kata – kata yang dianggap paling mendekati dan pada umumnya menggunakan pendekatan *Natural Language Processing* (NLP). *Natural Language Processing* merupakan salah satu tujuan jangka panjang dari *Artificial Intelligence* (kecerdasan buatan) yaitu pembuatan program yang memiliki kemampuan untuk memahami bahasa manusia. Bahasa yang digunakan dalam berkomunikasi atau berinteraksi berupa suara atau ucapan, tidak jarang juga ketika berinteraksi menggunakan bentuk tulisan.

Tidak lupa dalam pembuatan *chatbot* ada beberapa peran penting yaitu *training* dan *testing*. Didalam sebuah *training* terdapat peranan *tensorflow*, *tensorflow* adalah *open source library* untuk *machine learning* yang di release oleh Google yang mendukung beberapa bahasa pemrograman (Pratik Devikar, 2016). *Tensorflow* digunakan untuk melakukan eksperimen model *deep learning*, melatih model pada *dataseat* yang berukuran besar, dan membuatnya layak digunakan. Selain itu *Tensorflow* juga mendukung *training* dan *interface* bersekala besar dengan menggunakan ratusan server menggunakan *Graphic Processing Unit* (GPU) untuk *training* secara efisien.

Namun masih banyak perusahaan yang menggunakan pelayanan menggunakan *customer service* yang memiliki beberapa permasalahan yang sering dijumpai seperti keterbatasan dalam melayani seorang *customer*, kurangnya responsif dalam melayani berbagai pertanyaan *customer*. Setiap permasalahan pasti

memiliki efek samping yang akan berdampak seperti kehilangannya *customer* yang tetap (*loyal*), sehingga berpengaruh dalam naik turunnya suatu bisnis suatu perusahaan.

Oleh karena itu penting untuk mengembangkan suatu teknologi informasi dalam menghadapi hal tersebut. Pengembangan teknologi informasi tidak hanya dari manajemen sistem, tapi juga dapat diterapkan sampai pada hubungan pelanggan seperti mengembangkan suatu sistem kecerdasan buatan. Dalam pengembangannya terdapat contoh seperti ChatBot (Azwari, 2016), Question & Answering (Cristianto, 2015), chatbot monitoring (Wijaya, 2018), dan juga terdapat tipe chatbot *searching* (Satria, 2017). Dengan menggunakan teknologi berupa komputer dan menerapkan *Artificial Intelligence* didalamnya yang sangat berguna untuk mempermudah kebutuhan manusia saat ini. Seperti yang telah dijelaskan pada Al-qur'an pada surat alam nasyrah ayat 5 – 6

فَإِنَّ مَعَ الْعُسْرِ يُسْرًا (5) إِنَّ مَعَ الْعُسْرِ يُسْرًا (6)

” karena sesungguhnya sesudah kesulitan itu ada kemudahan. Sesungguhnya sesudah kesulitan itu ada kemudahan.”(QS. Al – Insyirah(94): ayat 5 – 6).

Dari Jalaluddin al – Mahalli & Jalaluddin as – Suyuthi, karena sesungguhnya sesudah kesulitan itu atau kesukaran itu ada kelapangan yakni kemudahan. Nabi sering mengalami kesulitan dan hambatan dari orang – orang kafir, kemudian beliau mendapatkan kelapangan dada dan kemudahan, yaitu setelah beliau mengalami kemenangan atas orang – orang kafir.

Dikarenakan semakin berkembangnya teknologi hingga saat ini maka tidaklah mungkin untuk mendapatkan berbagai kemudahan dari penggunaan

teknologi secara baik dan benar. Oleh karena itu penerapan *chatbot* dibutuhkan untuk mempermudah melayani pelanggan, mempermudah dalam pengelolaan pemesanan yang berguna untuk menaikkan hasil dalam melakukan bisnis. Untuk mengatasi hal tersebut maka diperlukan untuk melatih *chatbot* yang menghasilkan informasi lebih prediktif dan memiliki tingkat akurasi yang baik. Dengan penelitian ini diharapkan mampu memberikan solusi dari berbagai macam permasalahan saat ini.

1.2. Pernyataan Masalah

Berdasarkan latar belakang yang telah diuraikan sebelumnya maka didapatkan rumusan masalah yang diangkat adalah bagaimana mengimplemantasikan *Neural Network* pada *chatbot* untuk mengetahui seberapa akuratkah akurasi dan jawaban yang diberikan.

1.3. Tujuan Penelitian

Tujuan dari penelitian ini adalah untuk menerapkan metode *Neural Network* dan menunjukkan keakuratan dari metode tersebut pada aplikasi *chatbot*.

1.4. Batasan Masalah

Berdasarkan rumusan masalah tersebut, ruang lingkup penelitian dibatasi agar dapat menghasilkan penelitian yang lebih fokus, maksimal dan tidak meluas.

Ruang lingkup permasalahan akan dibatasi sebagai berikut :

1. Dialog dilakukan dalam Bahasa Indonesia yang sesuai dengan EYD (Ejaan Yang Disempurnakan).
2. Topik dialog dibatasi seputar informasi pemesanan travel.

1.5. Manfaat Penelitian

Dengan adanya penelitian ini diharapkan dapat memberikan manfaat sebagai berikut :

1. Dapat mempermudah *user* untuk melakukan pemesanan travel.
2. Bagi peneliti, diharapkan dapat mengetahui penerapan *Neural Network* pada chatbot.
3. Bagi peneliti, diharapkan dapat mengetahui kelebihan dan kekurangan dari *Neural Network* pada *chatbot*.

1.6. Sistematika Penulisan

Penelitian ini tersusun dalam laporan dengan terdiri dari beberapa bab pembahasan sebagai berikut :

Bab I Pendahuluan : Pada bab pertama ini berisi tentang latar belakang penelitian, identifikasi masalah, batasan masalah, tujuan penelitian, manfaat penelitian.

Bab II Tinjauan Pustaka : bab ini berisi tentang teori dasar yang digunakan sebagai referensi dalam pembuatan *chatbot*.

Bab III Analisis dan Perancangan : bab ini berisi penjabaran teori yang dipakai *chatbot*. Perancangan aplikasi terdiri atas perancangan proses – proses utama dan desain aplikasi yang terdiri atas desain output, desain input, desain *interface*.

Bab IV Hasil dan Pembahasan : bab ini berisi membahas tentang implementasi dari aplikasi *chatbot* yang mampu menjawab pertanyaan dari *user* yang membutuhkan informasi seputar pemesanan tiket. Disini juga dipaparkan pengujian terhadap aplikasi yang dibuat untuk mengetahui apakah

aplikasi tersebut dapat menyelesaikan masalah yang diberikan dengan jawaban yang sesuai.

Bab V Penutup : bab ini berisi tentang kesimpulan dan saran yang diharapkan dapat bermanfaat untuk pengembangan *chatbot* yang mampu berdialog kepada *user* menggunakan metode *Neural Network*

Daftar Pustaka : berisi daftar referensi yang tercantum dalam penelitian ini



BAB II

TINJAUAN PUSTAKA

Teori terkait yang dijelaskan dalam Tugas Akhir ini adalah teori dasar mengenai

2.1. Penelitian Terkait

Saat ini sudah terdapat berbagai penelitian yang membahas mengenai *chatbot* dengan pendekatan *Natural Lagunage Processing* (NLP) salah satunya oleh Elisabet Nila S.C.P dan Irawan Afrianto pada tahun 2015. Penelitian yang dilakukan bertujuan untuk mempermudah masyarakat untuk mendapatkan informasi seputar wisata kota Bandung. Dalam penilitian ini peneliti menggunakan teknik analisis pembuatan perangkat lunak dengan metode *waterfall*, yang meliputi beberapa proses diantaranya *system engineering, analysis, design, coding, implementation and testing*, dan *maintenance* (Ian Sommerville, 2009).

Yang kedua, terdapat jurnal yang membahas tentang *chatbot* yang digunakan untuk objek wisata Jawa Timur oleh Eka Larasati Amalia pada tahun 2017. Penelitian ini dilakukan untuk memaksimalkan pengembangan pencarian informasi seputar pariwisata di Jawa Timur dengan memanfaatkan *Arificial Intelligence Markup Language* (AIML). *Chatbot* ini berisi tentang informasi wisata, lokasi wisata, tiket masuk wisata, rute, dan fasilitas wisata. Penelitian ini menggunakan *Work Breakdown Structure* (WBS) untuk pengorganisasian sistemnya yang dimana agar data menjadi struktur yang lebih detail dalam tiap prosesnya maka dibuat *work breakdown structure*.

Yang ketiga, terdapat jurnal tentang menggantikan suatu peranan terhadap *customer service* dengan menggunakan metode *forward chaining* yang dibuat oleh

Ananda Dwi R, Firdha Imamah, Yusuf Mei Andre S, dan Ardiansyah pada tahun 2018. Penelitian yang dilakukan ini bertujuan untuk mengganti suatu peranan *customer service* yang dimana dengan keterbatasan dalam melayani pelanggan. Dalam pengumpulan data, peneliti tersebut melakukan wawancara terhadap pemilik UKM MINSU untuk mengetahui kinerja yang sudah dilalui hingga saat itu. Metode yang digunakan dalam pengembangan *chatbot* tersebut adalah dengan menggunakan *Extreme Programming* (XP), dalam pengerjaannya menggunakan *Extreme Programming* ada 4 kerangka yaitu *planning*, *design*, *coding*, dan *testing*.

Selanjutnya terdapat jurnal tentang implementasi *Natural Language Processing* (NLP) dengan konversi kata pada sistem *chatbot*. Jurnal ini dibuat oleh Mungki Astiningrum, Pramana Yoga Saputra, dan Maya Shoburu Rohmah pada tahun 2018. Di dalam jurnal ini implementasi dari *Levenshtein Distance* dan *TFIDF* digunakan untuk pembobotan suatu dokumen yang digunakan untuk mendapatkan nilai *cosine similarity* *Cosine similarity* dalam mencari jawaban yang sesuai dengan inputan user yaitu dengan membandingkan seluruh dokumen yang ada.

Selain itu terdapat jurnal tentang penerapan algoritma *Fulltext Search Boolean Mode* yang digunakan pada aplikasi *chatbot*. Algoritma ini merupakan salah satu fitur dari MySQL yang dimana digunakan untuk mencocokkan setiap data yang telah terdata sebelumnya. Cara kerja dari *Fulltext Search Boolean Mode* ini seperti query “SELECT LIKE”, perbedaan utamanya yaitu mampu memberikan dokumen yang paling relevan. Algoritma ini tidak berdiri sendiri dalam melakukan percakapan terhadap *user*, algoritma ini dibantu dengan adanya peran dari *Tf - idf* dan *program - o* sebagai bahan dasar *chatbot*. Penelitian ini dibuat oleh Nyoman Satria Paliwahet, I Made Sukarsa, dan I Ketut Gede Darma Putra pada tahun 2017.

Pada Tugas Akhir ini dibuat sebuah aplikasi *chatbot* seputar pemesanan tiket menggunakan metode *Neural Network*. Teori dasar yang digunakan pada tugas akhir ini seperti pada penelitian diatas.

2.2. Landasan Teori

2.2.1. Chatbot

Chatbot merupakan program komputer yang digunakan untuk mensimulasikan sebuah percakapan atau komunikasi yang interaktif oleh seseorang untuk membahas sesuatu. *Chatbot* disusun atas dua kata yaitu *chat* dan *bot*. *Chat* adalah komunikasi yang dilakukan melalui media tulisan atau pesan. *Bot* merupakan suatu program yang memiliki suatu pengetahuan yang dapat memberikan respon sesuai perintah yang telah diberikan. Tujuan pembuatan *chatbot* ini adalah untuk mengetahui apakah *chatbot* dapat menipu *user* agar mereka mengira sedang berkomunikasi dengan manusia (Vijayarani, Ilamathi dkk) .

Salah satu cara dalam pembuatan *chatbot* adalah dengan menggunakan *Natural Language Processing* (NLP) yang berisi sekumpulan pola (*pattern*) dan *template* yang digunakan *chatbot* untuk penelusuran jawaban dari kalimat yang masuk, NLP itu sendiri merupakan salah satu bidang ilmu Kecerdasan Buatan (*Artificial Intelligence*) yang mempelajari komunikasi antara manusia dengan komputer melalui bahasa alami (Ronen Feldman dan J.Sanger, 2007). Model komputasi seperti ini berguna untuk memudahkan komunikasi antara manusia dengan komputer dalam pencarian informasi, sehingga dapat terjadi suatu interaksi antara keduanya dengan menggunakan bahasa alami.

Chatbot dibangun sesuai dengan topik yang sudah dimodelkan dalam bisnis pengetahuan yang digunakan untuk memecahkan masalah seseorang untuk

keperluan pribadi maupun keperluan bisnis, sebagai contoh adalah *chatbot* yang menangani pemesana travel. Di dalam *chatbot* tersebut terdapat model pengetahuan untuk menjawab pertanyaan – pertanyaan yang sesuai dengan konteks yang telah disusun.

Chatbot terdiri dari tiga kombinasi, dimana ketiga kombinasi inilah yang membentuk sebuah *chatbot* (Ines Guzman, 2016), diantaranya adalah :

a) *User Interface*

User Interface dalam *chatbot* merupakan sebuah jembatan antara *chatbot* dan *user* saling berinteraksi melalui aplikasi berbasis *text*.

User Interface bertujuan untuk membuat *user* nyaman dan memudahkan ketika ingin menggunakan aplikasi.

b) *Artificial Intelligence*

AI atau *Artificial Intelligence* membuat *chatbot* mengerti dan memahami setiap interaksi yang terjadi dengan *user*. *Chatbot* menangani pemecahan masalah melalui aturan yang telah ditentukan sebelumnya.

c) Integrasi

Integrasi dengan sistem lainnya akan menambah kekayaan fitur yang terdapat didalam suatu *chatbot*. Dengan mengintegrasikan *chatbot* ke sistem yang lain dapat menyediakan informasi tambahan. Dengan cara ini *chatbot* mampu memberikan informasi yang lebih kaya kepada *user*. Di keperluan bisnis *chatbot* dapat membantu memecahan permasalahan yang berkaitan dengan komunikasi dengan pelanggan untuk meningkatkan pelayanan pengalaman dalam hal

berkomunikasi, *chatbot* sangat efektif digunakan dalam permasalahan yang fokus dan spesifik serta dapat diprediksi (Ines Guzman, 2016). Pada perkembangan dunia bisnis yang sangat cepat, *chatbot* menjadi salah satu alternatif solusi yang digunakan untuk memudahkan pelanggan menjangkau dan melakukan interaksi.

2.2.2. Preprocessing

Pada tahap *preprocessing* dilakukan penyiapan dokumen mentah menjadi dokumen atau representatif dokumen yang siap diproses untuk langkah selanjutnya (Zaman dan Winarko, 2011). Pada tahap ini proses yang dilakukan antara lain *case folding*, melakukan proses *tokenizing*, dan yang terakhir yaitu *stemming*.

a) Case Folding

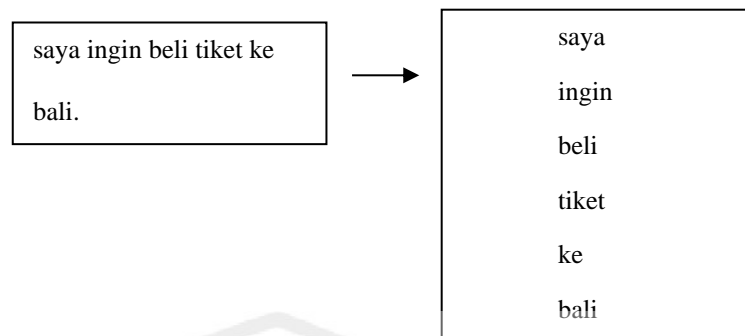
Case folding adalah mengubah semua huruf dalam dokumen menjadi huruf kecil. Hanya huruf 'a' sampai dengan 'z' yang diterima. Karakter selain huruf akan dihilangkan dan dianggap sebagai delimiter (Ronen Feldman, 2007). Berikut contoh dari *case folding*.



Gambar 2.1 Contoh *Case Folding*

b) Tokenizing

Pada dasarnya proses *tokenizing* merupakan proses memisahkan setiap kata yang menyusun suatu dokumen. Umumnya setiap kata teridentifikasi atau terpisahkan oleh karakter spasi, sehingga proses *tokenizing* mengandalkan karakter spasi pada sebuah dokumen untuk melakukan pemisahan kata. Berikut contoh dari *tokenizing*.

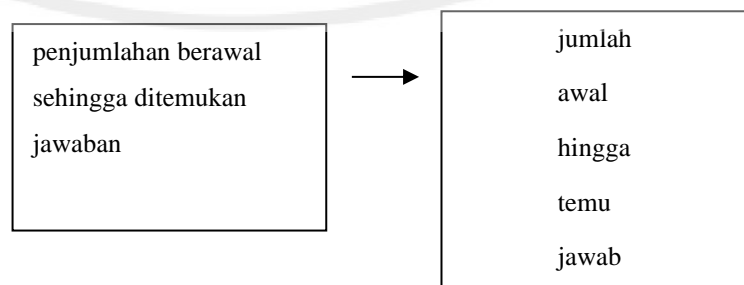


Gambar 2.2 Contoh *Tokenizing*

c) *Stemming*

Stemming merupakan suatu proses dimana setiap kata akan dikembalikan menjadi kata dasar. *Stemming* teks berbahasa Indonesia memiliki permasalahan yang sangat khusus, masalah tersebut merupakan perbedaan tipe – tipe dari imbuhan (*affixes*). Imbuhan (*affixes*) dapat berubah sesuai dengan huruf kata dasar, ketika dalam suatu kata terdapat lebih dari satu imbuhan (*affixes*) yang melekat pada suatu kata maka urutan dari dalam menghilangkan kata imbuhan tersebut menjadi sangat penting apabila urutan yang dilakukan salah maka kata dasar yang benar dari kata tersebut tidak akan ditemukan. Untuk menghilangkan imbuhan yang berada diakhir (*suffixes*) terlebih dahulu untuk menghilangkan imbuhan yang berada diawal (*prefix*).

Berikut contoh dari *stemming*.

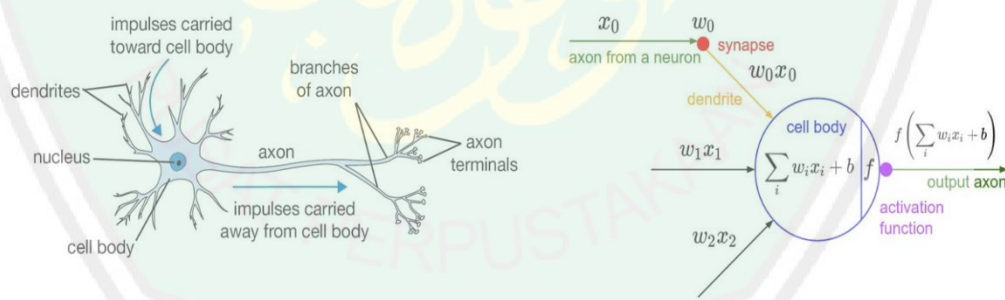


Gambar 2.3 Contoh *Stemming*

2.2.3. Neural Network

Neural network mengadopsi dari cara kerja dari kemampuan otak manusia yang mampu memberikan stimulasi / rangsangan, melakukan proses, dan memberikan output. Output diperoleh dari berbagai macam variasi stimulasi dan proses yang terjadi di dalam otak manusia. Ide dasar dari *neural network* itu sendiri dimulai dari otak manusia, dimana otak manusia memuat sekitar 10^{11} neuron. Neuron yang berfungsi sebagai tempat memproses informasi yang masuk.

Dari struktur neuron pada otak manusia maka konsep dasar dari *neural network* terbentuk. Ide mendasar dari *neural network* adalah mengadopsi mekanisme berpikir sebuah sistem atau aplikasi yang mengikuti cara kerja otak, baik untuk pemrosesan berbagai sinyal elemen yang diterima dan toleransi terhadap kesalahan atau *error*. Pada Gambar 2.3 Terdapat ilustrasi bagaimana penggambaran kemiripan Neural Network dengan neuron otak.

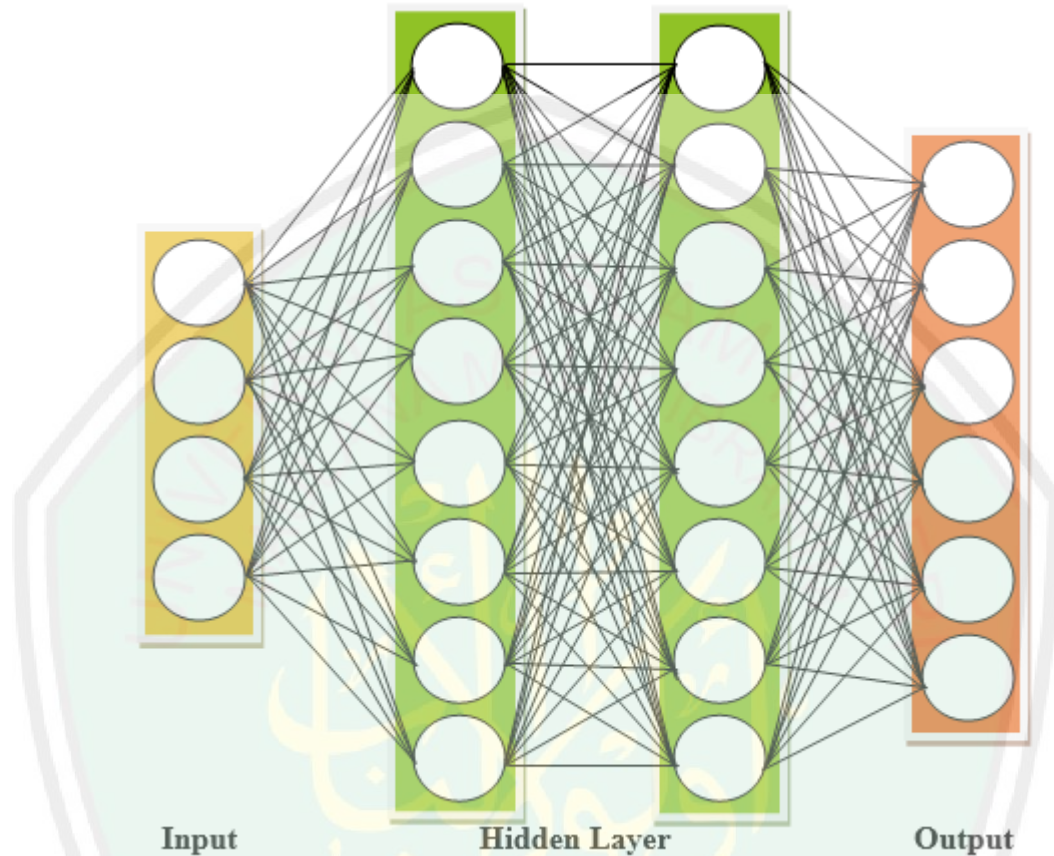


Gambar 2.4 Ilustrasi Kemiripan Neural Network dan Neuron Otak.

Sumber: (Samuelsena., 2017)

Neural network dibangun dari banyak node atau unit yang dihubungkan oleh *link* secara langsung. *Link* yang digunakan dalam *neural network* berfungsi sebagai propagasi aktivitas dari unit satu ke unit lainnya. *Neural network* memiliki sejumlah

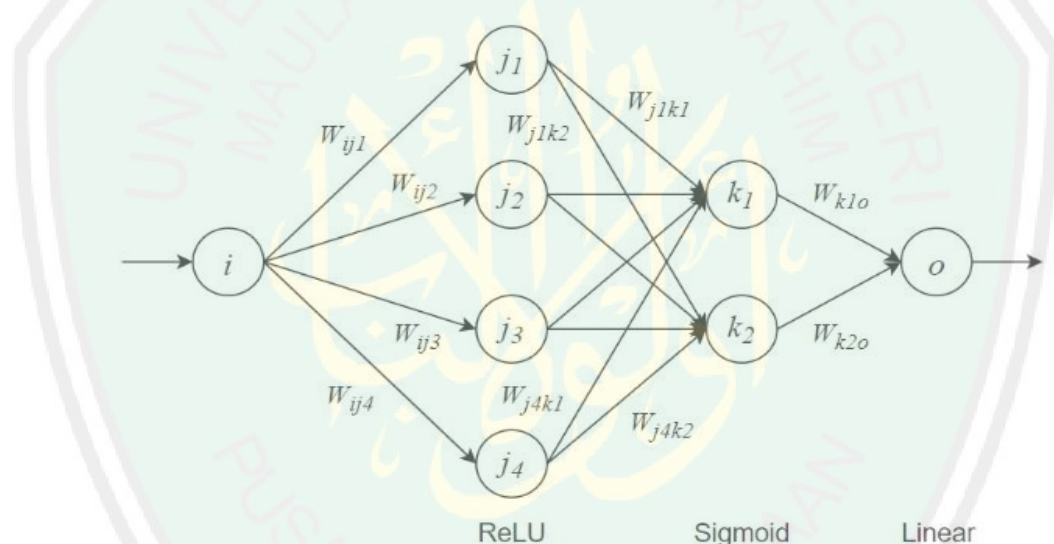
prosesor sangat sederhana dan saling berhubungan yang disebut sebagai *neuron*. *Neuron* yang terhubung dengan pembobotan (*weight*) melewati sinyal dari *neuron* satu ke *neuron* lainnya seperti gambar dibawah ini.



Gambar 2.5 Neural Network 2 Hidden Layer

Penelitian ini menggunakan klasifikasi *feed forward* menggunakan 2 *hidden layer* dan 8 *bias* tiap *hidden layer*nya dengan tahap pembelajaran *back propagation*, *backpropagation* merupakan salah satu bagian dari *neural network* yang mempunyai target dalam hal pencarian target dengan meminimalikan *error* pada *output* yang dihasilkan. Penggunaan *neural network* dalam pembuatan *chatbot* merupakan yang paling efektif diantara lainnya, seperti yang tertera pada jurnal sebelumnya penggunaan AIML (*Artificial Intelligence Modeling Language*) pada *chatbot* hanya menghasilkan akurasi dari 80% hingga 90% dari setiap hasilnya

dengan catatan pertanyaan yang ditanyakan oleh *user* sudah berada di *list* pertanyaan yang telah disediakan berbeda dengan penggunaan *tf-idf* akurasi yang dihasilkan mencapai 90% keatas dengan catatan apabila nilai dari *tf-idf* berbeda sedikit maka akurasi yang akan dihasilkan jauh berada dibawah 60%. Dari hasil tersebut untuk melakukan *training* dan *testing*, *neural network* membutuhkan waktu yang relatif lama, cepat lambatnya waktu yang diperlukan ditentukan dari spesifikasi komputer yang digunakan dan dibutuhkan 4000 langkah atau *epoch* sesuai standar untuk mendapatkan hasil yang maksimal. Berikut contoh perhitungan akurasi dan data *loss* menggunakan *forward pass*.



Gambar 2.6 Model Sederhana *Forward Pass*. Sumber: (Samuelsena., 2017)

Neural network menggunakan 2 *hidden layer* dengan *hidden layer* pertama menggunakan ReLU, *hidden layer* kedua menggunakan sigmoid, dan terakhir *output* menggunakan linear sebagai *activation function*. Dari gambar di atas didapati 4 *weight* dan 4 bias diantara *input layer* dan *hidden layer* pertama, kemudian 8 *weight* dan 2 bias diantara *hidden layer 2* dan *output*, dengan

i = *input* (nilai kata yang dirubah menjadi angka)

w = *weight* (pembobotan nilai)

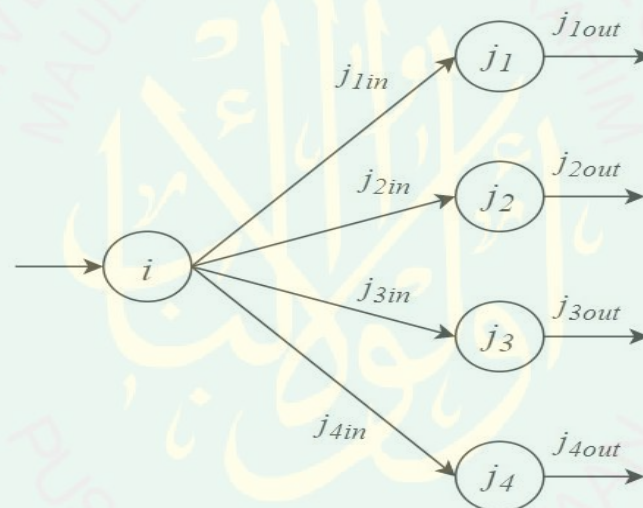
j = nilai pada layer pertama hasil perhitungan *input* dan *weight*

menggunakan perhitungan ReLU

k = nilai pada layer kedua hasil perhitungan dari layer pertama dan *weight*

bias = penyesuaian nilai pada *weight* yang digunakan untuk *input* dan *output*

Selanjutnya ketika sudah menentukan nilai *weight* dan *bias*, selanjutnya yaitu melakukan perhitungan antara *input* dan *hidden layer 1*.



Gambar 2.7 Proses Training Input ke Layer Pertama. Sumber: (Samuelsena., 2017)

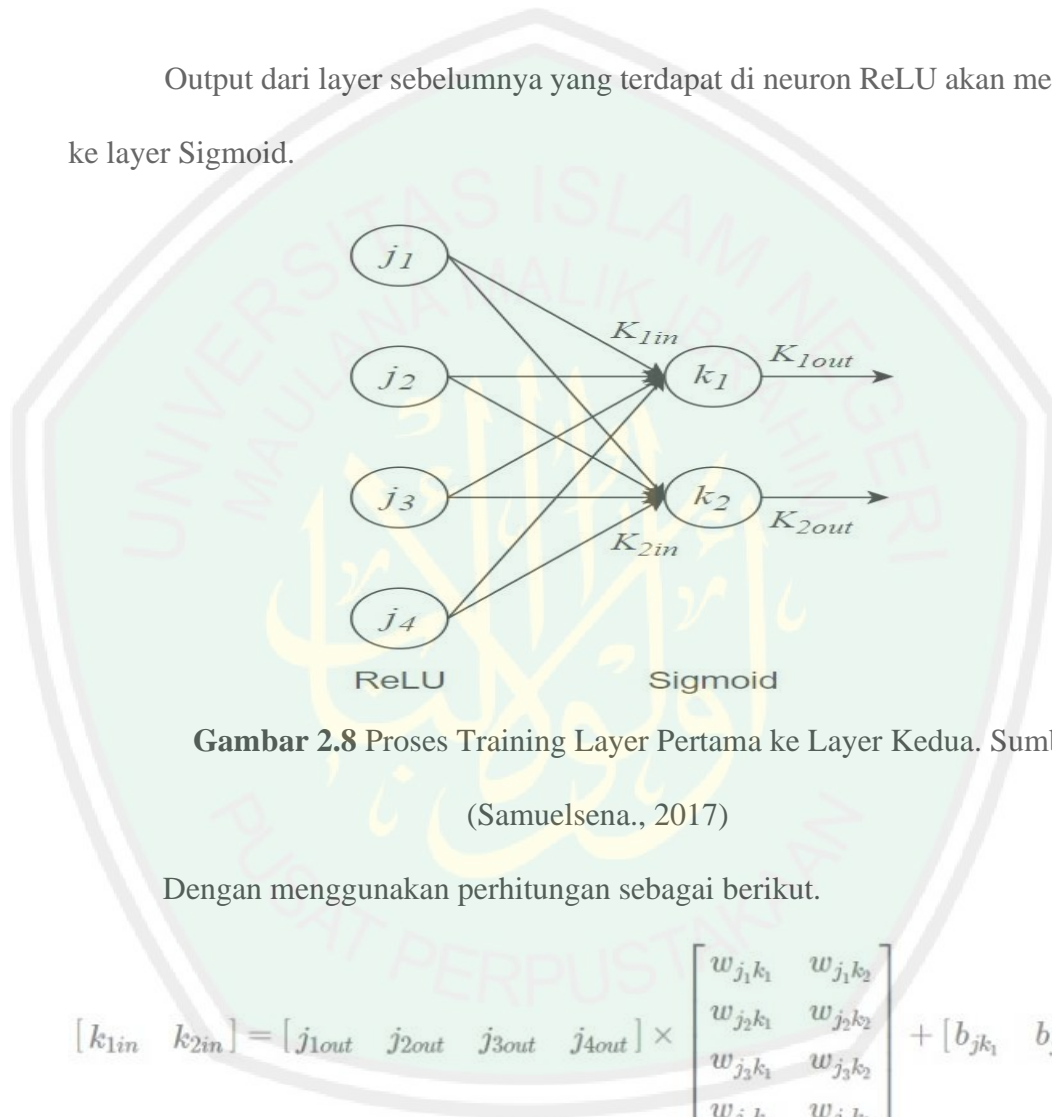
Hal pertama yang dilakukan adalah dengan melakukan perkalian antara *dot product* dan penjumlahan matrix antara *input*, *weight*, dan *bias*. Dengan perhitungan sebagai berikut.

$$[j_{1in} \ j_{2in} \ j_{3in} \ j_{4in}] = [input] \times [w_{ij_1} \ w_{ij_2} \ w_{ij_3} \ w_{ij_4}] + [b_{ij_1} \ b_{ij_2} \ b_{ij_3} \ b_{ij_4}] \quad (2.1)$$

Ketika *input* dari setiap node pada *hidden layer 1* memiliki nilai, semua nilai tersebut akan dikeluarkan setelah melalui *activation function*. Pada *hidden layer 1 activation function* yang digunakan sebagai berikut.

$$\text{ReLU}([j_{1in} \ j_{2in} \ j_{3in} \ j_{4in}]) = [\max(0, j_{1in}) \ \max(0, j_{2in}) \ \max(0, j_{3in}) \ \max(0, j_{4in})] \quad (2.2)$$

Output dari layer sebelumnya yang terdapat di neuron ReLU akan mengalir ke layer Sigmoid.



Gambar 2.8 Proses Training Layer Pertama ke Layer Kedua. Sumber:

(Samuelsena., 2017)

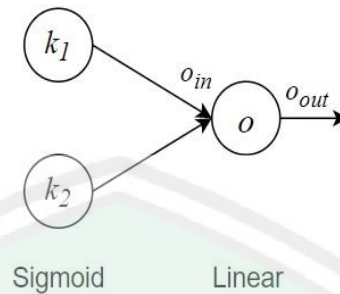
Dengan menggunakan perhitungan sebagai berikut.

$$[k_{1in} \ k_{2in}] = [j_{1out} \ j_{2out} \ j_{3out} \ j_{4out}] \times \begin{bmatrix} w_{j_1k_1} & w_{j_1k_2} \\ w_{j_2k_1} & w_{j_2k_2} \\ w_{j_3k_1} & w_{j_3k_2} \\ w_{j_4k_1} & w_{j_4k_2} \end{bmatrix} + [b_{jk_1} \ b_{jk_2}]$$

Setelah perhitungan tersebut telah selesai maka selanjutnya menghitung berdasarkan *activation function*.

$$\text{Sigmoid} \Rightarrow f(x) = \frac{1}{1 + e^{-x}} \quad (2.3)$$

Berikutnya yang terakhir adalah perhitungan diantara *hidden layer 2* dan *output*, hasil dari *hidden layer 2* akan diteruskan ke *linear layer (output)*.



Gambar 2.9 Proses Training Layer Kedua ke Output. Sumber:

(Samuelsena., 2017)

Dengan menggunakan perhitungan sebagai berikut.

$$[o_{in}] = [k_{1out} \quad k_{2out}] \times \begin{bmatrix} w_{k_1 o} \\ w_{k_2 o} \end{bmatrix} + [b_o] \quad (2.4)$$

Pada *linear layer*, *activation function* yang digunakan untuk mencari prediksi jawaban sebagai berikut.

$$Linear \Rightarrow f(x) = x \quad (2.5)$$

Setelah perhitungan sudah mencapai *linear layout*, langkah berikutnya yaitu dengan menghitung data *error*.

$$Loss = \frac{1}{2} (Prediction - Target)^2 \quad (2.6)$$

$$Loss = \frac{1}{2} (o_{out} - output)^2$$

BAB III

METODOLOGI PENELITIAN

3.1. Objek Penelitian

Pada penelitian ini akan dikembangkan sebuah aplikasi *chatbot* berbasis web untuk melakukan pemesanan travel atau tiket menggunakan metode *Neural Network*.

3.2. Desain Aplikasi

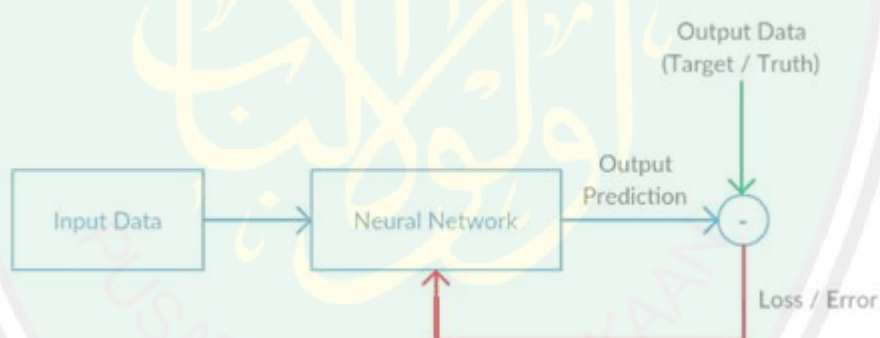
Pada bab ini akan dibahas mengenai aplikasi yang akan dibangun. Aplikasi yang dibangun merupakan sebuah aplikasi *chatbot* yang membahas tentang pemesanan travel. Aplikasi ini berbasis *web*, sehingga *user* dapat mengakses dimanapun dan kapanpun mereka berada.

Aplikasi ini dirancang dan dibangun secara bertahap, tahapan pertama yaitu *input* yang dimasukan *user* atau pertanyaan yang akan ditanyakan oleh *user*. Kemudian setelah *user* memberikan pertanyaan, kalimat tersebut akan memasuki tahap *preprocessing*. *Preprocessing* merupakan suatu tahapan awal dalam mengelola data input sebelum memasuki proses utamanya. *Preprocessing* dilakukan dengan bertujuan untuk penyeragaman dan kemudahan pembacaan ketika telah memasuki proses utamanya. *Preprocessing* sendiri terdiri dari *case folding*, *tokenizing*, dan *stemming*. *Case folding* yaitu proses mengubah huruf yang telah dimasukan oleh *user* menjadi huruf kecil. Proses *tokenizing* merupakan pemecahan kalimat dengan menghilangkan tanda baca dan simbol lainnya. Pemecahan kalimat dilakukan berdasarkan spasi dari tiap kata hingga didapatkan potongan kata. Langkah selanjutnya yaitu *stemming* digunakan untuk menghilangkan kata imbuhan dan menjadikannya menjadi kata dasar.

Ketika proses *preprocessing* sudah selesai, selanjutnya *chatbot* tersebut akan melakukan *training*. Dalam melakukan *training chatbot*, didalamnya ada beberapa proses yang dilakukan terlebih dahulu salah satunya perubahan kata menjadi angka tensor menggunakan *tensorflow* dikarenakan dalam proses tersebut tidak dapat memahami data struktur kata. Setelah dilakukan *training* proses selanjutnya yaitu *testing* yang dimana *chatbot* tersebut akan dilakukan beberapa kali uji coba untuk mendapatkan jawaban dari pertanyaan *user*. Didalam proses tersebut akan didapatkan grafik pelatihan dari setiap uji coba dan jawaban yang diberikan untuk *user* adalah jawaban yang mirip atau hampir mendekati.

3.3. Desain Proses

Berdasarkan dari desain aplikasi didapatkan alur proses dari pembuatan *chatbot*, berikut alur dari proses tersebut.



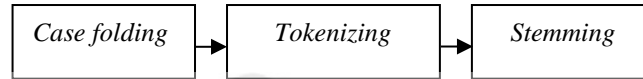
Gambar 3.1 Alur Desain Proses. Sumber: (Samuelsena., 2017)

3.3.1. Input Kalimat

Kalimat yang digunakan untuk melakukan percakapan *chatbot* ini menggunakan kalimat yang EYD. Dari setiap kalimat akan diproses lebih lanjut untuk mendapatkan hasil secara maksimal dengan melakukan proses *preprocessing* agar mendapatkan jawaban yang sesuai dari pertanyaan yang telah diajukan.

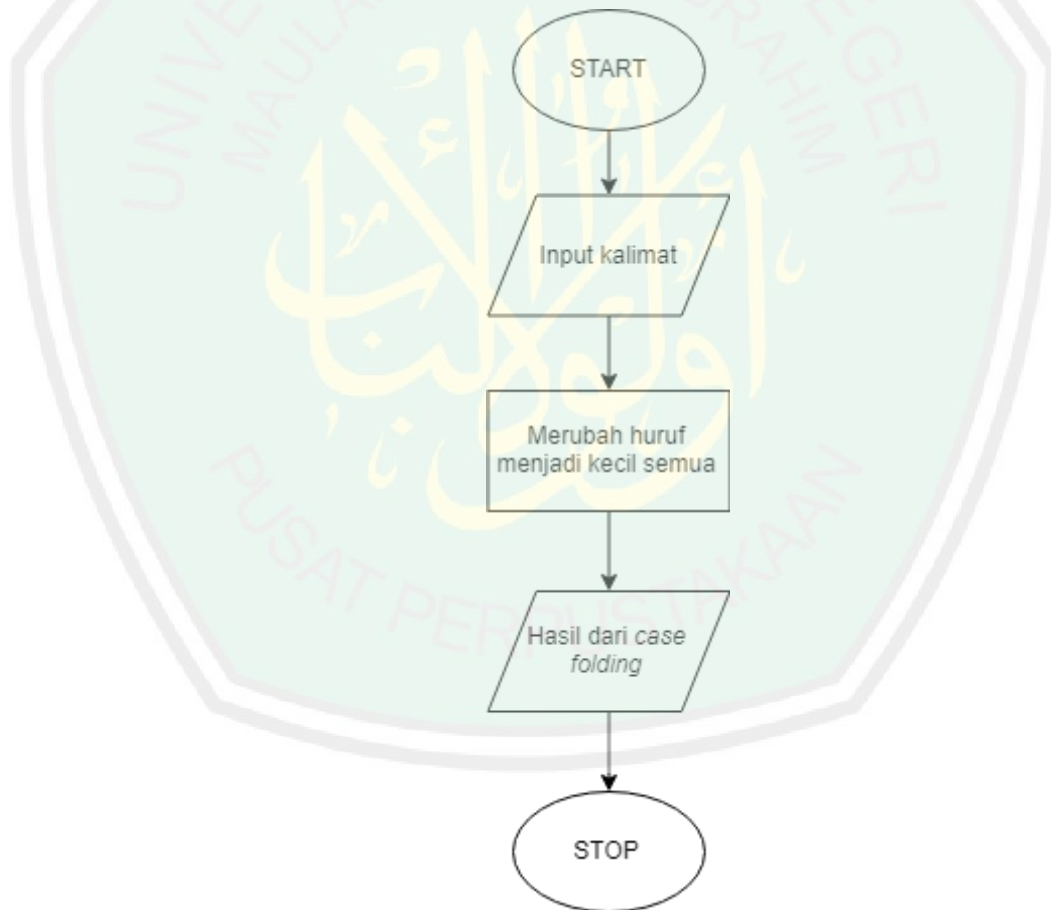
3.3.2.Preprocessing

Dalam pembuatan *chatbot* setelah adanya pertanyaan dari *user* langkah selanjutnya yaitu proses *preprocessing*, berikut alur proses *preprocessing* tersebut.



Gambar 3.2 Alur *Preprocessing*

Pada tahap *preprocessing* dalam penelitian ini meliputi *case folding*, *tokenizing* dan *stemming*. Proses *case folding* menggunakan fungsi `.lower()`. Berikut alur dari *case folding*.

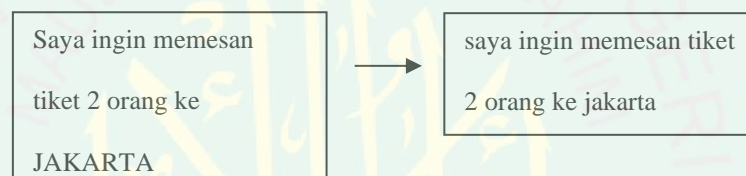


Gambar 3.3 Proses *Case Folding*

Keterangan :

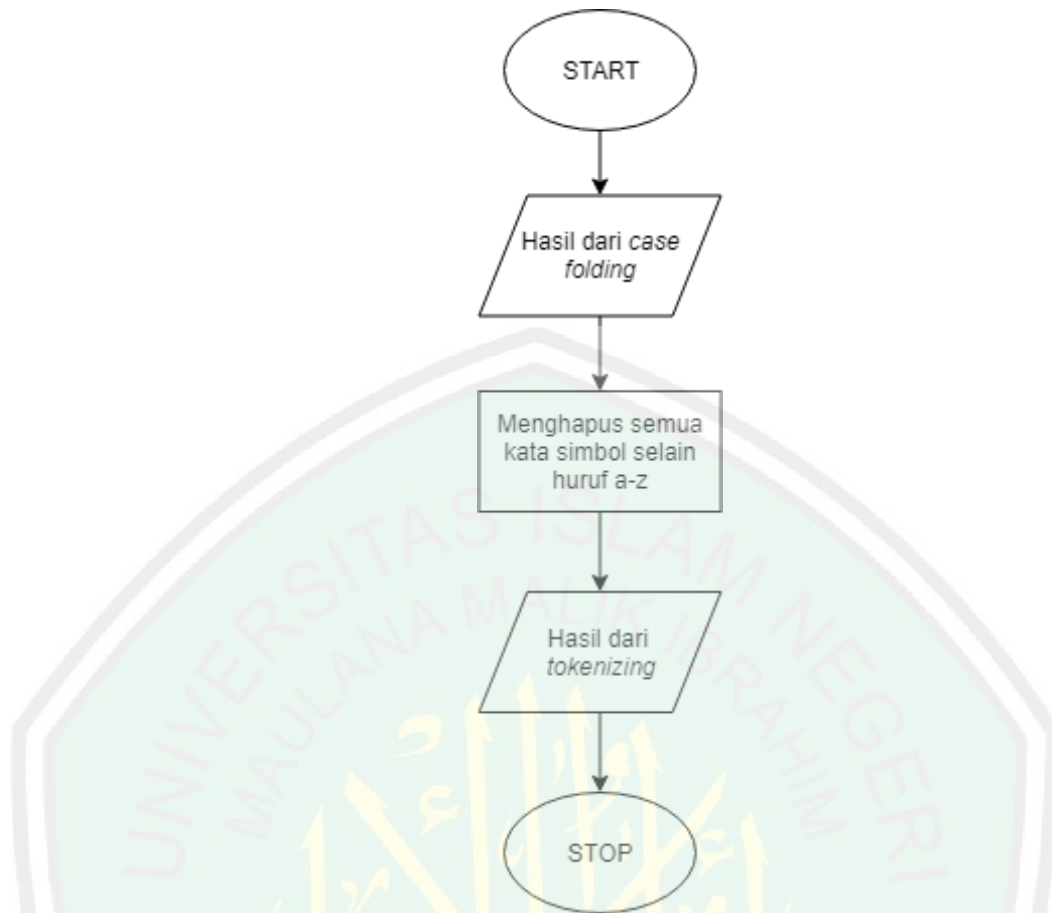
1. Memulai program.
2. *User* memasukan *input* pertanyaan.
3. Setelah *input* masuk, perintah **.lower()** akan merubah huruf menjadi kecil semua.
4. Output dari *case folding* berupa pengecilan huruf yang masuk.
5. Selesai.

Setelah proses tersebut selesai, hasil outputnya bisa dilihat pada gambar dibawah ini.



Gambar 3.4 Hasil *Case Folding*

Dilanjutkan dengan melakukan pemisahan kata berdasarkan spasi yang kemudian dimasukan kedalam sebuah tabel untuk proses selanjutnya, proses tersebut dinamakan *tokenizing*. Berikut alur dari *tokenizing*.

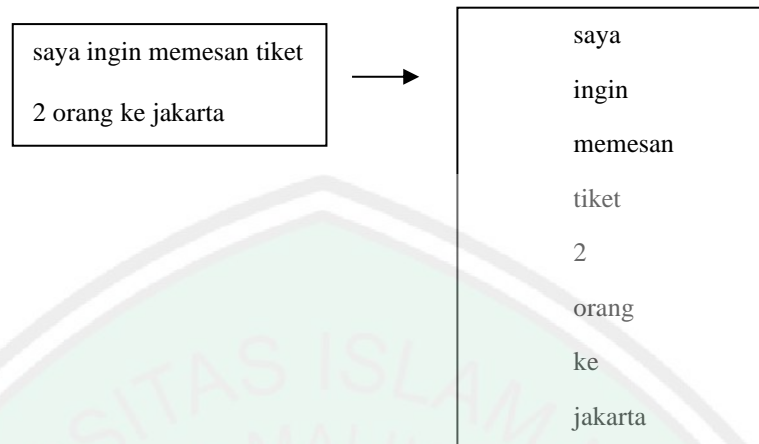


Gambar 3.5 Proses *Tokenizing*

Keterangan :

1. Memulai program.
2. Hasil output dari *case folding* akan diolah kembali untuk proses selanjutnya.
3. Setelah output dari *case folding* masuk maka kata – kata tersebut akan dipisahkan berdasarkan spasi.
4. Output dari *tokenizing* berupa sekumpulan kata yang terdiri dari beberapa tabel.
5. Selesai.

Setelah proses tersebut selesai, hasil outputnya bisa dilihat pada gambar dibawah ini.



Gambar 3.6 Hasil *Tokenizing*

Proses *stemming* adalah proses dimana setiap kalimat atau kata yang memiliki kata imbuhan akan dihilangkan semua termasuk kata imbuhan yang berada didepan, ditengah, dibelakang, dan kalimat yang memiliki imbuhan baik didepan dan diakhir, berikut alur dari *stemming* sebagai berikut.

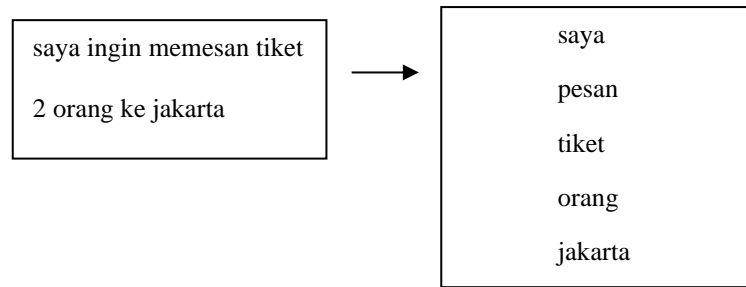


Gambar 3.7 Proses *Stemming*

Keterangan :

1. Memulai program
2. Hasil dari *tokenizing* akan diolah kembali untuk proses selanjutnya.
3. Ketika tahap *tokenizing* selesai maka selanjutnya setiap kata akan dihilangkan kata imbuhan yang terletak di depan, dibelakangan, dan ditengah.
4. Output dari *stemming* merupakan bentuk kata dasar dari kata tersebut.
5. Selesai.

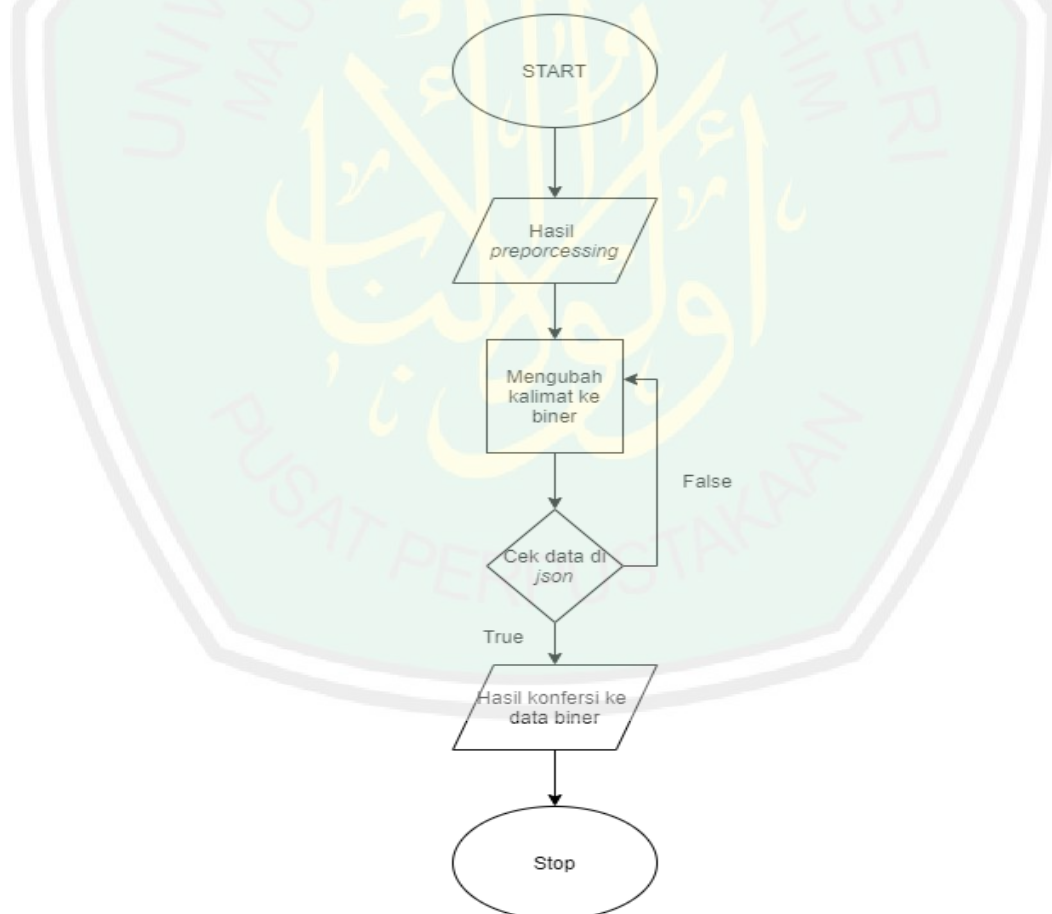
Setelah proses tersebut selesai, hasil outputnya bisa dilihat pada gambar dibawah ini.



Gambar 3.8 Hasil *Stemming*

3.3.3. Training Data

Dalam sebuah *chatbot* untuk mendapatkan jawaban dari pertanyaan *user* maka *chatbot* tersebut diperlukan suatu *training*, *training* yang dimaksud disini adalah suatu proses megelolah bagaimana dari satu pertanyaan dapat menghasilkan jawaban yang sesuai, berikut alur dari pelatihan data.



Gambar 3.9 Proses Perubahan Angka

Keterangan :

1. Memulai program.
2. Hasil dari *preprocessing* diolah kembali untuk untuk dilakukan pengecekan setiap kata apabila memiliki kesamaan didalam *corpus*.
3. Apabila ditemukan kesamaan kata dalam *intern pattern* dan *intern class* maka kata tersebut diubah menjadi 1 dan apabila kata tidak ditemukan dalam *corpus* akan dirubah menjadi 0. Bila dalam satu kata terdapat perulangan kata maka dihitng sebagai satu kata.
4. Selesai.

Setelah proses tersebut sudah selesai, hasil *output* yang dihasilkan berupa satu buah array yang berisikan angka, angka tersebut merupakan *intern pattern* dan *intern class* yang ditetapkan sebagai *train_x* dan *train_y*.

```
train_x example: [0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 1, 0, 0, 0, 1]
train_y example: [0, 0, 1, 0, 0, 0, 0, 0, 0, 0]
```

Gambar 3.10 Contoh Perubahan Angka

Setelah semua proses *training* selesai, selanjutnya adalah proses melakukan uji coba didalam sebuah layer. Alur proses dapat dilihat pada Gambar 3.1, berikut adalah tahapan dari uji coba, yaitu :

a) Input

Input merupakan data yang diperoleh dari memberikan *questioner* ke beberapa masyarakat yang sering berpergian menggunakan travel. Data yang masuk akan dimasukan kedalam suatu file berbentuk Json yang

nantinya setiap pertanyaan akan dilatih sebagai model dasar dari chatbot.

```

{"intents": [
  {"tag": "exc",
   "patterns": ["?oocnnkkk"],
   "responses": ["Mohon maaf, silahkan bertanya lagi"]
  },
  {"tag": "salam",
   "patterns": ["Hai", "Hallo", "Halo", "Salam", "Hello", "Hei", "hi", "Mau bertanya", "Permisi"],
   "responses": ["Hallo, ada yang bisa dibantu?", "Hai ada yang bisa dibantu?"],
   "context_set": ""
  },
  {"tag": "salam2",
   "patterns": ["Assalamualaikum", "Assalamu'alaikum", "Assalamu'alaikum Warahmatullahi", "Assalamu'alaikum Warahmatullahi Wabarakatuh", "Assalamualaikum", "Assalamu'al"],
   "responses": ["Waalaikumussalam"],
   "context_set": ""
  },
  {"tag": "salam3",
   "patterns": ["Selamat pagi", "Pagi", "Met pagi", "Pegeee", "Morning", "Good Morning"],
   "responses": ["Selamat pagi"],
   "context_set": ""
  },
  {"tag": "salam4",
   "patterns": ["Selamat siang", "siang", "Met siang", "Good afternoon"],
   "responses": ["Selamat siang"],
   "context_set": ""
  },
  {"tag": "salam5",
   "patterns": ["Selamat malam", "malam", "Met malam", "Malem", "Good evening", "ninght", "Good night"],
   "responses": ["Selamat malam"],
   "context_set": ""
  },
  {"tag": "perpisahan",
   "patterns": ["Bye", "Dada", "selamat tinggal", "sampai jumpa", "senang bisa berjumpa", "senang bisa bertemu", "c ya", "see ya", "see you", "see u"],
   "responses": ["Byee, Terima kasih sudah berkunjung"]
  },
  {"tag": "terimakasih",
   "patterns": ["Terima kasih", "thanks", "Makasih", "maaci", "trim", "oke", "tq"],
   "responses": ["Senang bisa membantu"]
  },
  {"tag": "jam",
   "patterns": ["Kapan nelawan buka?", "Berana lama nelawan buka?", "Samnai kapan nelawan berlangsund", "Jam herana bukanwa?", "untuk hari travel dimana", "Ana

```

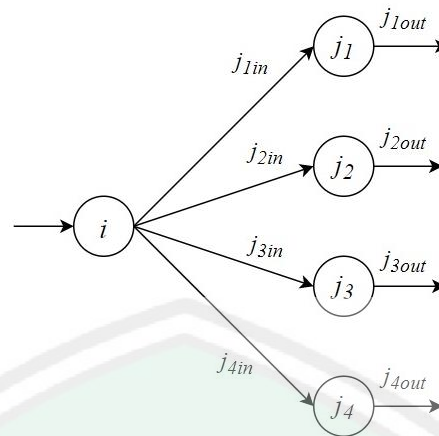
Gambar 3.11 Data dalam Bentuk File Json

b) Word Embedding

Input kalimat akan mengalami perubahan dari kata menjadi angka seperti pada Gambar 3.6, setelah perubahan proses selanjutnya seperti pada Gambar 3.5.

c) Proses pelatihan menggunakan *Deep Neural Network*

Pelatihan yang dilakukan menggunakan *Deep Neural Network* dengan tahapan pelatihan berjumlah 3500 *steps* menggunakan proses *Feed forward*. Dimana dalam setiap tahapan pelatihan akan menampilkan akurasi dan data *loss*. Proses pertama dari *input* ke layer pertama dapat dilihat pada Gambar 3.7.



Gambar 3.12 Training *Input* ke Layer Pertama. Sumber:

(Samuelsena., 2017)

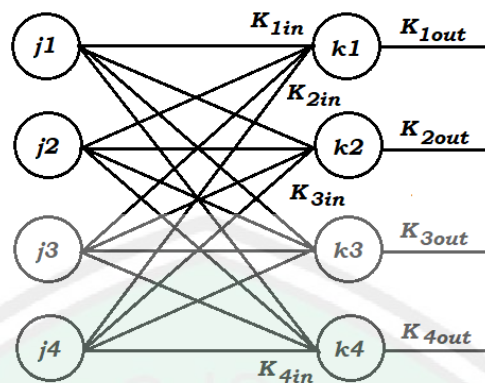
Input merupakan hasil dari *word embedding* yang kemudian akan dilakukan perhitungan berdasarkan ReLU. ReLU adalah cara paling dasar untuk memperkenalkan non – linearitas kedalam jaringan saraf.

Menghitung nilai dari input ke layer pertama :

$$\begin{aligned} [j_{1in} \ j_{2in} \ j_{3in} \ j_{4in}] &= [input] \times [w_{ij_1} \ w_{ij_2} \ w_{ij_3} \ w_{ij_4}] + [b_{ij_1} \ b_{ij_2} \ b_{ij_3} \ b_{ij_4}] \\ &= [1] \times [1.0 \ 1.0 \ 1.0 \ 1.0] + [1.0 \ 1.0 \ 1.0 \ 1.0] \\ &= [2.0 \ 2.0 \ 2.0 \ 2.0] \end{aligned}$$

$$\begin{aligned} ReLU([j_{1in} \ j_{2in} \ j_{3in} \ j_{4in}]) &= [\max(0, j_{1in}) \ \max(0, j_{2in}) \ \max(0, j_{3in}) \ \max(0, j_{4in})] \\ &= [\max(0, 2.0) \ \max(0, 2.0) \ \max(0, 2.0) \ \max(0, 2.0)] \\ &= [2.0 \ 2.0 \ 2.0 \ 2.0] \end{aligned}$$

Output tersebut akan diteruskan ke *layer* kedua seperti pada Gambar 3.8.



Gambar 3.13 Training Layer Pertama ke Layer Kedua. Sumber:

(Samuelsena., 2017)

Untuk mendapatkan nilai dari neuron di layer kedua digunakan rumus sebagai berikut.

Menghitung nilai layer kedua :

$$[k_{1in} \ k_{2in} \ k_{3in} \ k_{4in}] = [j_{1out} \ j_{2out} \ j_{3out} \ j_{4out}] \times \begin{bmatrix} w_{j_1k_1} & w_{j_1k_2} & w_{j_1k_3} & w_{j_1k_4} \\ w_{j_2k_1} & w_{j_2k_2} & w_{j_2k_3} & w_{j_2k_4} \\ w_{j_3k_1} & w_{j_3k_2} & w_{j_3k_3} & w_{j_3k_4} \\ w_{j_4k_1} & w_{j_4k_2} & w_{j_4k_3} & w_{j_4k_4} \end{bmatrix} + [b_{j_1k_1} \ b_{j_1k_2} \ b_{j_1k_3} \ b_{j_1k_4}]$$

$$= [2.0 \ 2.0 \ 2.0 \ 2.0] \times \begin{vmatrix} 0.25 & 0 & 0.75 & 0.75 \\ 0.75 & 0.25 & 0 & 0.5 \\ 0.5 & 0.5 & 0.25 & 0.25 \\ 0.75 & 0.75 & 0.5 & 0.5 \end{vmatrix} + [1.0 \ 1.0 \ 1.0 \ 1.0]$$

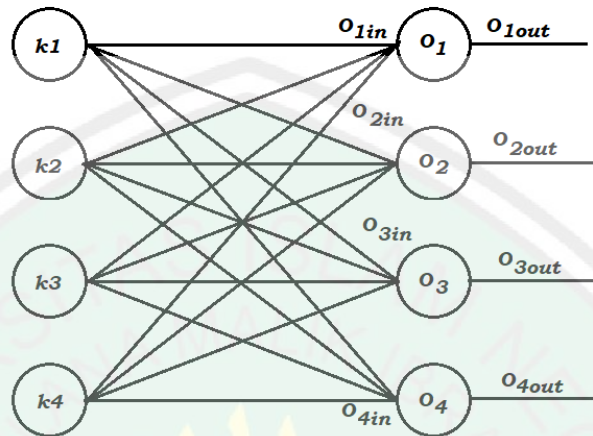
$$= [6.0 \ 5.0 \ 4.0 \ 5.0]$$

$$\text{Sigmoid} \Rightarrow f(x) = \frac{1}{1 + e^{-x}}$$

$$\text{Sigmoid}([k_{1in} \ k_{2in} \ k_{3in} \ k_{4in}]) = \left[\frac{1}{1+e^{-k_{1in}}} \quad \frac{1}{1+e^{-k_{2in}}} \quad \frac{1}{1+e^{-k_{3in}}} \quad \frac{1}{1+e^{-k_{4in}}} \right]$$

$$= [0.9975 \ 0.9933 \ 0.9891 \ 0.9933]$$

Sigmoid merupakan proses untuk memperbarui setiap bobot yang ada dari setiap neuron. Hasil dari neuron kedua akan diteruskan ke linear layer seperti gambar dibawah ini.



Gambar 3.14 Training Layer Kedua ke Linear Layer. Sumber:

(Samuelsena., 2017)

Untuk mendapatkan nilai dari linear layer digunakan rumus seperti dibawah ini.

Menghitung nilai layer output :

$$\begin{aligned}
 [o_{in}] &= [k_{1out} \quad k_{2out} \quad k_{3out} \quad k_{4out}] \times \begin{bmatrix} w_{k_1o} \\ w_{k_2o} \\ w_{k_3o} \\ w_{k_4o} \end{bmatrix} + [b_1 \quad b_2 \quad b_3 \quad b_4] \\
 &= [0.9975 \quad 0.9933 \quad 0.9891 \quad 0.9933] \times \begin{array}{c} 1.0 \\ 0.5 \\ 0.25 \\ 0.5 \end{array} + [1.0 \quad 1.0 \quad 1.0 \quad 1.0] \\
 &= 3.733
 \end{aligned}$$

Setelah mendapatkan output pada neuron pertama pada linear layer selanjutnya yaitu menghitung data *loss*, berikut perhitungannya.

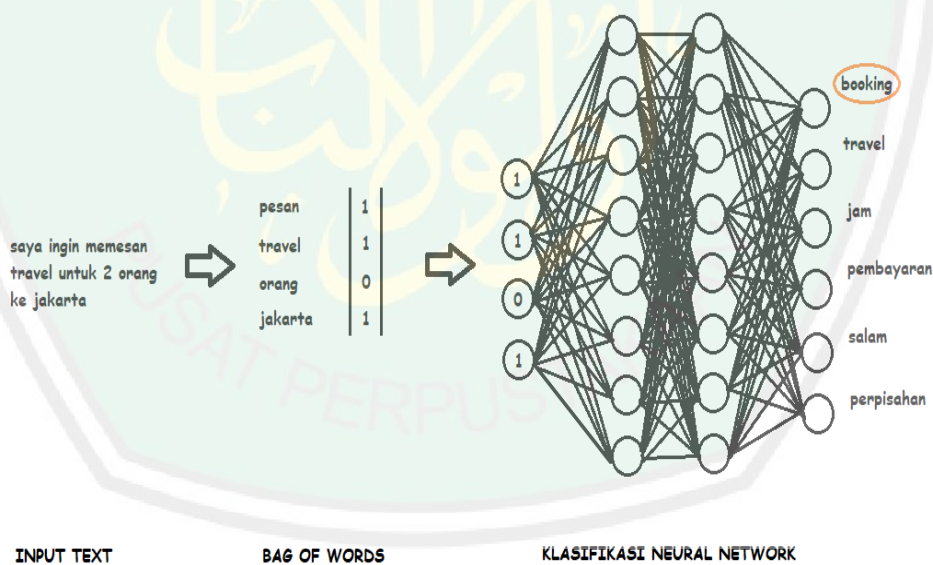
$$Loss = \frac{1}{2}(Prediction - Target)^2$$

$$Loss = \frac{1}{2}(o_{out} - output)^2$$

$$= \frac{1}{2}(3.733 - 3)^2$$

$$= 0.268$$

Error yang didapatkan akan dilatih terus hingga 5000 *steps* seperti pada Gambar 2.5 hingga mendapati nilai *error* yang kecil, *error* dan prediksi tersebut merupakan salah satu bentuk pencarian jawaban berdasarkan pelatihan yang dilakukan. Pada gambar 3.10 merupakan gambaran sederhana pelatihan *neural network*.



Gambar 3.15 Model Training Neural Network

Pertanyaan – pertanyaan yang telah dikumpulkan dimasukkan kedalam kelompok untuk dibedakan macam – macam pertanyaannya. Dapat dilihat pada **Lampiran 1**. Setiap pertanyaan tersebut disimpan dalam file berbentuk *json*,

pertanyaan yang sudah dikumpulkan akan dilatih untuk mencari kesamaan dari pertanyaan dari user yang kemudian memberikan jawaban berdasarkan kecocokan yang didapatkan. Pertama pertanyaan yang masuk akan dirubah menjadi angka dan setiap kata yang memiliki kecocokan atau kemiripan dengan kata kunci yang berada didalam file *json* akan mendapatkan nilai 1 sedangkan bila tidak maka bernilai 0. Proses perubahan dapat dilihat seperti Gambar 3.16.

tolong saya ingin pesan tiket untuk 2 orang

```
data y : [[0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0],
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1], [0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
[0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0], [0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
[1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
[0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0], [0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0]]
```

Gambar 3.16 Perubahan Nilai

Model training tersebut akan disimpan kedalam file *tflern* yang digunakan untuk mempermudah proses selanjutnya. Semakin banyak data model maka semakin bagus output yang nantinya akan didapatkan, berikut potongan program dalam melakukan penyimpanan *training* model.

Pengecekan pada *data y* merupakan pencocokan kategori satu kalimat berdasarkan kata kunci yang berada dalam file *json*. Kemudian dari hasil yang didapatkan akan dilakukan proses pelatihan dengan tahapan 3500 *steps* dan pelatihan menggunakan sampel data dari 4 angka pertama hingga ke *final layer* untuk mendapatkan nilai prediksi dari pertanyaan *user*. Nilai prediksi kemudian akan dibandingkan dengan label yang telah ditentukan untuk memperbarui nilai bobot pada *final layer* melalui proses *back-propagation*. Tiap pelatihan akan mendapatkan nilai akurasi dan data loss, yang dimana semakin rendah nilai *loss* yang didapatkan ketika pelatihan maka semakin baik hasilnya. Proses pelatihan hingga ke langkah 5000 dapat dilihat seperti Gambar 3.17

```

[A][A] Training Step: 45495 | total loss: 0.00062 | time: 0.020s
[K] Adam | epoch: 3500 | loss: 0.00062 - acc: 1.0000 -- iter: 32/50
[A][A] Training Step: 45496 | total loss: 0.00058 | time: 0.020s
[K] Adam | epoch: 3500 | loss: 0.00058 - acc: 1.0000 -- iter: 36/50
[A][A] Training Step: 45497 | total loss: 0.00062 | time: 0.022s
[K] Adam | epoch: 3500 | loss: 0.00062 - acc: 1.0000 -- iter: 40/50
[A][A] Training Step: 45498 | total loss: 0.00056 | time: 0.022s
[K] Adam | epoch: 3500 | loss: 0.00056 - acc: 1.0000 -- iter: 44/50
[A][A] Training Step: 45499 | total loss: 0.00052 | time: 0.022s
[K] Adam | epoch: 3500 | loss: 0.00052 - acc: 1.0000 -- iter: 48/50
[A][A] Training Step: 45500 | total loss: 0.00047 | time: 0.031s
[K] Adam | epoch: 3500 | loss: 0.00047 - acc: 1.0000 -- iter: 50/50

```

Gambar 3.17 Proses Pelatihan

Pelatihan tersebut akan disimpan kedalam suatu file yang nantinya akan digunakan sebagai dasar dari model jawaban.

```

# build neural network
net = tflearn.input_data(shape=[None, len(train_x[0])]) #input
net = tflearn.fully_connected(net, 8) # layer
net = tflearn.fully_connected(net, 8) #layer
net = tflearn.fully_connected(net, len(train_y[0]), activation = 'softmax') #output
net = tflearn.regression(net) #regresi

# pemodelan dan setup tensorboard
model = tflearn.DNN(net, tensorboard_dir='tflearn_logs')
#callbacks = TensorBoard(log_dir='./Graph')
file_writer = tf.summary.FileWriter('tflearn_logs', net.graph)

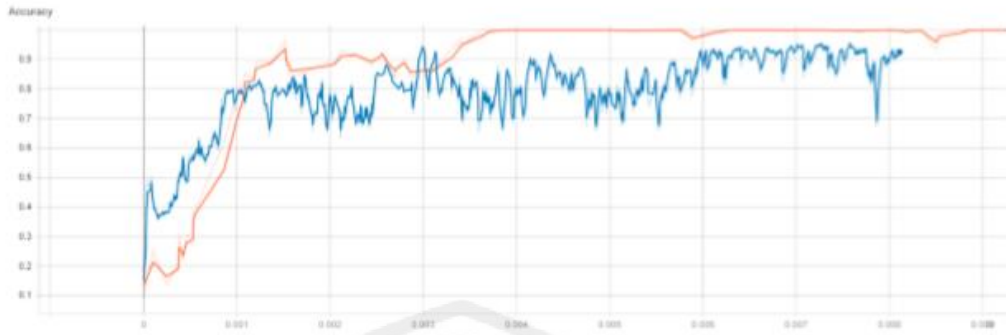
# start training
model.fit(train_x, train_y, n_epoch=5000, batch_size=32, show_metric=True)
##tf.summary.add_graph(net.graph)
model.save('model.tflearn')

#save model data
pickle.dump( {'words':words, 'classes':classes, 'train_x':train_x, 'train_y':train_y}, open( "training_data", "wb" ) )

```

Gambar 3.18 Penyimpanan Model Training

Hasil dari pelatihan model akan berbentuk grafik yang menunjukkan setiap melakukan pelatihan dan banyaknya data yang ada akan membuat chatbot semakin baik dalam memberikan jawaban ke *user*, berikut contoh grafik yang didapatkan ketika melakukan suatu pelatihan.



Gambar 3.19 Grafik Akurasi Pelatihan

Pelatihan akan dilakukan secara berulang kali hingga *steps* yang ditentukan dengan menggunakan nilai parameter pada Tabel 3.1.

Tabel 3.1 Nilai Parameter Proses Pelatihan

Parameter	Nilai parameter
Epoch	3500
Iteration	50
Training Step	45500
Batch Size	4

Dan didapatkan akurasi seperti pada Tabel 3.2.

Tabel 3.2 Akurasi yang Didapatkan

Epoch	Error Target	Akurasi
1000	0.01	65%
2500	0.01	76%
5000	0.01	90%

3.3.4. Output

Tahap terakhir dari semua proses yaitu output yang berisikan jawaban yang berdasarkan dari pertanyaan dari *user*. Sebelum mendapatkan hasil dari klasifikasi ada beberapa proses yang harus dilewati yaitu dengan melakukan ranking dari setiap pertanyaan yang masuk kemudian diurutkan dari ranking yang tinggi hingga rendah berdasarkan kecocokannya terhadap *corpus* yang ada, model training yang sebelumnya yang telah disimpan akan dipanggil ke dalam model pengujian, kemudian akan dilakukan print out jawaban dari chatbot seperti pada gambar dibawah ini.

```

model.load('./model.tflearn')

context = {}

ERROR_THRESHOLD = 0.25
def classify(sentence) :
    # probabilitas
    results = model.predict([bow(sentence, words)])[0]
    # filter prediksi
    results = [(i, r) for i, r in enumerate(results) if r > ERROR_THRESHOLD]
    # ranking probabilitas
    results.sort(key = lambda x: x[1], reverse = True)
    return_list = []
    for r in results :
        return_list.append((classes[r[0]], r[1]))
    return return_list
    # print(model.predict([p]))

def response(sentence, userID = '123', show_details = False) :
    results = classify(sentence)
    # pencarian persamaan tag
    if results :
        # loop bila ada
        while results :
            for i in intents['intents'] :
                # cari kecocokan tag
                if i['tag'] == results[0][0] :

                    if 'context_set' in i :
                        if show_details : print ('context :', i['context_set'])
                        context[userID] = i['context_set']

                    if not i !=('intents'):
                        print("Mohon maaf, silahkan bertanya lagi")

                    if not 'context_filter' in i or \
                        (userID in context and 'context_filter' in i and i ['context_filter'] == context[userID]) :
                        if show_details : print ('tag :', i['tag'])
                        # randpm respon
                        # print("identify")
                        # return print(random.choice(i['responses']))
                        result_response = random.choice(i['responses'])
                        print(result_response)
                        return result_response

    results.pop(0)

```

Gambar 3.20 Kode Program Rangking Jawaban

Dalam memberikan jawaban, *chatbot* memberikan jawaban berdasarkan ranking tertinggi atau akurasi yang tinggi untuk dijadikan jawaban dari pertanyaan

user. Dari hasil pengujian yang telah dilakukan, ada faktor – faktor yang mempengaruhi akurasi seperti kemiripan kata, perbedaan bahasa, dan juga pertanyaan yang kurang lengkap. Meskipun memiliki kata yang berbeda namun memiliki makna yang sama hal tersebut dapat dikenali kemiripannya dengan tingkat akurasi yang cukup baik. Berbeda dengan gaya bahasa yang diucapkan sehari – hari yang membuat akurasi tidak sesuai ekspektasi namun memiliki hasil yang sama. Ketika pertanyaan dari *user* tidak lengkap atau hanya sebatas satu kata maka satu kata tersebut menjadi kunci dalam pencocokan jawaban yang terbilang memiliki akurasi yang rendah, bila pertanyaan *user* tidak dapat dijawab ada 2 hal yang menjadi faktornya yaitu pertanyaan yang kurang lengkap atau pertanyaan tersebut memang tidak tersedia, pertanyaan dari *user* tersebut akan disimpan dan akan dilakukan pengujian lagi setelahnya. Dapat dilihat perbedaan yang didapatkan akurasi pada Gambar 3.19.



Gambar 3.21 Perbedaan Penggunaan Kata

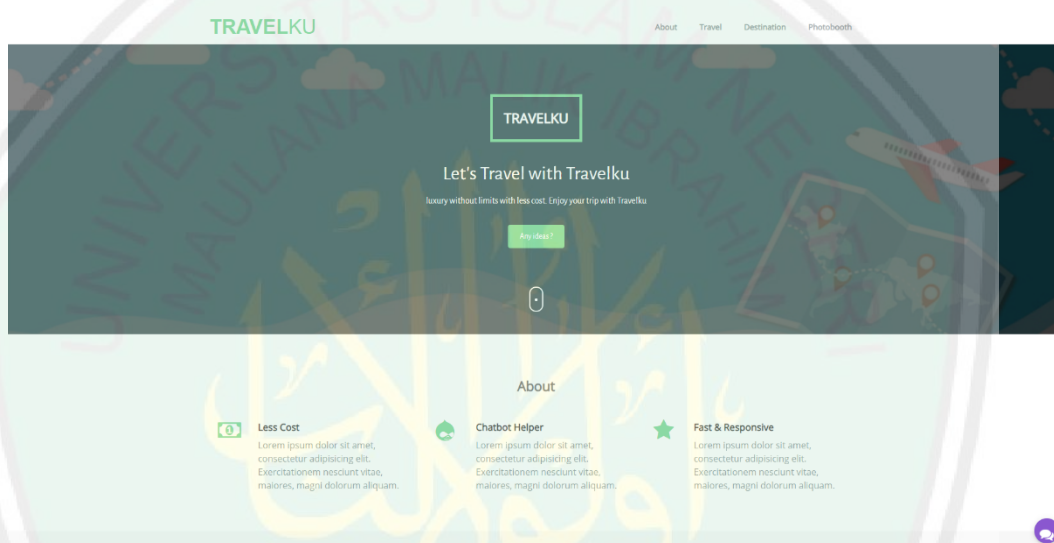
3.4 Implementasi Desain Antarmuka

Aplikasi web yang bertemakan travel ini, dibangun dengan menggunakan bahasa pemrograman *python* dan *framework flask*. Pembuatan antarmuka

menggunakan HTML, Javascript, dan JQuery. Dalam memperbagus tampilan web menggunakan CSS3.

3.4.1 Halaman Beranda

Pada halaman beranda ini hanya menampilkan data seperti halaman pada travel biasanya yang menampilkan foto, jenis – jenis travel, dan berbagai macam info lainnya yang. Halaman beranda dapat dilihat pada Gambar 3.14.



Gambar 3.22 Halaman Utama

Pada bagian pojok bawah kanan terdapat fitur chatbot untuk melakukan percakapan dengan bot yang bersangkutan dengan pembahasan travel.

3.4.1 ChatBot Messenger

Fungsi chatbot pada web ini digunakan untuk menjawab pertanyaan *user* yang dimana pertanyaan – pertanyaan tersebut harus bersangkutan dengan travel dan pertanyaan menggunakan Bahasa Indonesia sesuai dengan EYD. Untuk melakukan pengenalan chatbot tersebut menggunakan Flask sebagai jembatan untuk dapat diakses di website, berikut potongan kode programnya.

```

s=0
app = Flask(__name__, template_folder='./templates',static_folder='./templates/static')

def insert_sql(user_input):
    global s
    s=s+1
    resp = str(response(user_input))
    try:
        sql = 'insert into user_bot_chat values("' +str(s)+'","'+user_input+'","'+resp+'");'
        a.execute(sql)
        conn.commit()
    except:
        print("error")

def user_list():
    user=[]
    sql = 'select User_input from user_bot_chat;'
    a.execute(sql)
    w_user=list(a.fetchall())
    for i in w_user:
        user.append('User : '+i[0])
    return(user)

def bot_list():
    bot=[]
    sql = 'select Bot_output from user_bot_chat;'
    a.execute(sql)
    w_bot=list(a.fetchall())
    for i in w_bot:
        bot.append('Bot : '+i[0])
    return(bot)

```

Gambar 3.23 Potongan Kode Program List Pertanyaan

Potongan kode tersebut berfungsi untuk menyimpan sementara pertanyaan *user* kedalam database, tujuan dari disimpan pertanyaan tersebut digunakan sebagai perulangan untuk dapat melihat pertanyaan dan jawaban sebelumnya. Kemudian setelah disimpan dalam bentuk *list*, *list* tersebut dipanggil untuk dilakukan pencocokan jawaban dari hasil klasifikasi. Berikut potongan kode program dapat dilihat pada gambar dibawah ini.


```

def r():
    try:
        user_input = request.form['user_input']
        insert_sql(user_input)
        r=[]
        user = user_list()
        bot = bot_list()
        for j in range(0,len(user)):
            r.append(user[j])
            r.append(bot[j])
        return(r)
    except:
        r=[]
        user = user_list()
        bot = bot_list()
        for j in range(0,len(user)):
            r.append(user[j])
            r.append(bot[j])
        return(r)

@app.route('/')
def index():
    return render_template('index.html')

@app.route('/process',methods=['POST'])
def process():
    # inp = request.form['inp']
    # res = response(inp)
    # # response = str(response)
    # print(res)
    # return render_template('index.html',inp=inp,)
    return render_template('index.html', user_input=r())

if __name__ == '__main__':
    try:
        conn= pymysql.connect(host='localhost', user='root', password='root', db='chat')
        a= conn.cursor()
    except:
        print("connection error")
    app.run(host='127.10.0.0', debug=True, port=int('8000'))
    # app.run(port=5002, debug=True)
    conn.close()
    a.close()

```

Gambar 3.24 Potongan Kode Program Pencocokan Pertanyaan

Pertanyaan dan jawaban yang disimpan menggunakan MySQL sebagai databasenya, dengan isi database user_input sebagai input yang masuk dari user dan bot_output sebagai jawaban dari chatbot. Dapat dilihat seperti pada gambar dibawah ini.

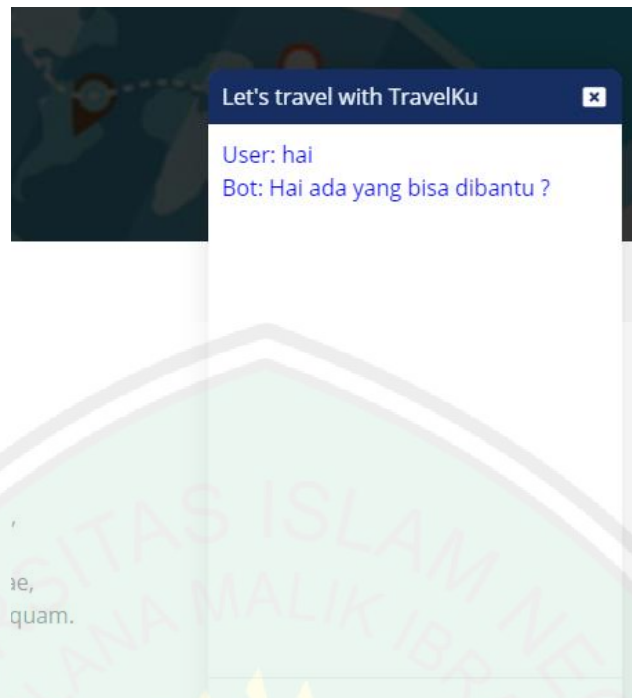
```

mysql> select * from user_bot_chat;
+-----+-----+-----+
| S_no | User_input | Bot_output |
+-----+-----+-----+

```

Gambar 3.25 Database Penyimpanan Sementara

Kemudian setelah disimpan akan ditampilkan di website yang terdapat fitur chatbotnya, dapat dilihat pada gambar dibawah ini.



Gambar 3.26 Tampilan Chatbot

Apabila pertanyaan tidak sesuai dan tidak menggunakan Bahasa Indonesia yang sesuai dengan EYD, maka chatbot akan menampilkan *None* dengan artian pertanyaan tidak bersangkutan dengan travel atau penggunaan bahasa yang tidak sesuai.

BAB IV

UJI COBA DAN PEMBAHASAN

Bab ini akan membahas tentang hasil dari implementasi yang telah dibahas dalam Bab 3 untuk menjawab setiap pertanyaan user sesuai dengan perancangan sistem sebelumnya.

4.1 Skenario Pengujian

Langkah pertama yang dilakukan untuk melakukan pengujian yaitu pengumpulan pertanyaan seputar *travelling* yang didapatkan dari beragam pertanyaan dari internet serta pengumpulan pertanyaan dari orang sekitar. Proses training data menggunakan bahasa pemrograman python berbasis *web flask* untuk python. Proses perancangan tersebut menggunakan *hardware* dan *software* sebagai berikut.

4.1.1. Spesifikasi *Hardware*

Spesifikasi perangkat keras yang digunakan dalam pembuatan *chatbot* ini adalah :

1. Processor Core i7-7700HQ 2.80GHz
2. 8.0 GB RAM
3. VGA Nvidia Geforce GTX 1060 6 GB
4. SSD NVMe 120 GB

4.1.2. Spesifikasi *Software*

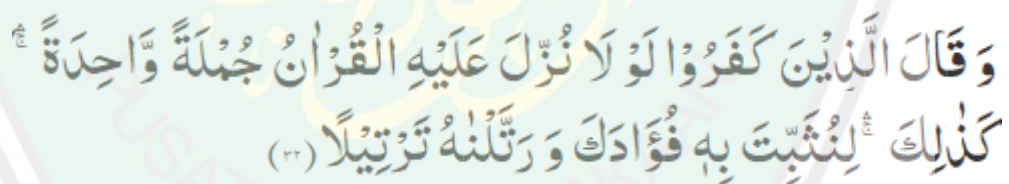
Spesifikasi *software* yang digunakan dalam pembuatan *chatbot* ini adalah:

1. Sistem Operasi Windows 10 64-bit
2. Visual Studio Code 1.42.1
3. Python 3.7

4. MySql
5. Browser Google Chrome

4.2 Pengujian Sistem

Pada tahap pengujian ini akan dilakukan proses tanya jawab seputar *travelling*, dengan ketentuan menggunakan Bahasa Indonesia yang sesuai dengan EYD. Hasil pengujian ini menggunakan *feedforward* neural network dengan tahapan pembelajaran *backpropogation* seperti yang telah dibahas di Bab 2 dan dapat dilihat di Tabel 4.3 yang menunjukkan hasil uji coba pertanyaan yang diberikan. Didalam tabel tersebut terdapat kolom pertanyaan yang menunjukan pertanyaan yang diajukan oleh user, kolom berikutnya yaitu jawaban yang merupakan jawaban dari hasil pencocokan kata yang sesuai dengan pertanyaan, kemudian terdapat kolom akurasi yang menunjukan hasil akurasi yang dihasilkan dari pelatihan. Dapat dilihat pada **Lampiran 2**. Dalam pelatihan sistem tersebut telah dijelaskan pada surat Al – Furqan ayat 32:



 وَقَالَ الَّذِينَ كَفَرُوا لَوْلَا نُزِّلَ عَلَيْهِ الْقُرْآنُ جُمْلَةً وَّاحِدَةً

 كَذَلِكَ لِنُثَبِّتَ بِهِ فُؤَادَكَ وَرَتَّلْنَاهُ تَرْتِيلاً (٣٢)

Dan orang – orang kafir berkata : “*Mengapa Al – Qur’an itu tidak diturunkan kepadanya sekali turun saja ?*” Demikianlah, agar Kami memperteguh *hatimu (Muhammad) dengannya dan Kami membacakannya secara tartil (berangsur – ansur, perlahan dan benar)*. (QS. Al – Furqan (25): Ayat 32).

Menurut Tafsir Al – Wajiz, Al – Qur’an itu tidak diturunkan secara sekaligus, tetapi diturunkan secara berangsur – ansur dengan begitu hati Nabi Muhammad shallallah ‘alaihi wa sallam menjadi kuat, tenang dan teguh. Terutama

ketika ada sebab – sebab gelisah, maka dengan diturunkannya Al – qur’an dapat menentramkannya.

Dari Tafsir Al – Mukhtashar, Dan orang – orang kafir berkata, “Mengapa Al – qur’an tidak diturunkan kepada Muhammad sekaligus seperti Taurat, Injil, dan Zabur ?” Allah berfirman, “Demikianlah Kami menurunkannya secara berangsur – angsur supaya Kami perkuat hatimu dengannya wahai Rasul dan Kami menurunkannya sedikit demi sedikit agar mudah dibaca dan dipahami.

Zabdatut Tafsir Min Fathil Qadir, orang – orang kafir itu berkata: “Mengapa Al – Qur’an tidak diturunkan kepada Muhammad sekaligus sebagaimana Taurat diturunkan kepada Musa ?” Lalu Allah membalas mereka. “Kami menurunkannya kepadamu secara terpisah dan melalui cara ini, sungguh akan menguatkan hatimu dan memudahkanmu dalam menjaga dan memahaminya. Kami juga membacaknya melalui lisan Jibril dengan sangat bagus dan perlahan – lahan.”

Dari Jalaluddin al – Mahalli dan Jalaluddin as – Suyuthi, berkatalah orang – orang yang kafir, “Mengapa tidak diturunkan kepadanya Al-qur’an sekali turun saja ? sebagaimana kitab Taurat, kitab Injil dan kitab Zabur.” Allah menjawab melalui firmanNya, “Kami sengaja menurunkannya demikian secara terpisah – pisah supaya Kami perkuat hatimu dengannya, Kami menguatkan kalbumu dengan Al-qur’an dan Kami membacaknya kelompok demi kelompok. Kami menurunkannya tahap demi tahap secara perlahan dan tidak tergesa – gesa, supaya mudah dipahami dan dihafal.

Berangsur – angsur yang dimaksud disini bila dikolerasikan dengan *Neural Network* terdapat keterkaitan yaitu pada pelatihan *chatbot* yang dimana dimulai dari training yang terus dilatih satu persatu yang kemudian diulang kembali untuk

mendapatkan hasil yang maksimal hingga akhirnya mendapatkan nilai kecocokan atau nilai akurasi yang tinggi.

Berdasarkan **Lampiran 2** menunjukkan bahwa pengujian menghasilkan rata – rata memberikan jawaban atas pertanyaan user sebesar 84% dengan pengujian akurasi seperti pada Gambar 4.1.



Gambar 4.1 Akurasi Pengujian yang Digunakan

Hasil dari pemodelan Neural Network yang digunakan dalam *chatbot* untuk setiap klasifikasi didapatkan seperti pada Tabel 4.1.

Tabel 4.1 Pertanyaan yang Sering Diajukan *User*

Orang ke	Klasifikasi %							
	Salam	pembayaran	jam	travel	fasilitas	tujuan	pembelian	harga
1	92	85	88	-	82	94	90	85
2	70	90	90	-	75	78	73	83
3	84	85	92	-	-	84	85	80
4	-	75	84	90	90	82	75	92
5	-	80	72	75	92	70	85	70

Dari Tabel 4.4 didapatkan hasil dari uji coba kepada *user* yang mencoba melakukan tanya jawab dengan *bot* Travelku. Dari sekian pertanyaan yang tersedia,

pertanyaan yang sering ditanyakan dimulai dari jadwal, harga, fasilitas, tujuan dll. Rata – rata akurasi yang didapatkan 83%, dari akurasi yang didapatkan jawaban dari setiap pertanyaan *user* telah terjawab dengan akurat, yang membedakan dari setiap akurasi yang didapatkan hanya dari perbedaan cara penulisan *user* ketika melakukan sesi tanya jawab terhadap *bot*. Dari hasil jawaban yang diberikan dilakukannya klasifikasi untuk menentukan tingkat akurasinya dapat dilihat pada Gambar 4.2

```
def classify(sentence) :
    # probabilitas
    results = model.predict([bow(sentence, words)])[0]
    # filter prediksi
    results = [[i , r] for i , r in enumerate(results) if r > ERROR_THRESHOLD]
    # ranking probabilitas
    results.sort(key = lambda x: x[1], reverse = True)
    return_list = []
    for r in results :
        return_list.append((classes[r[0]], r[1]))
    return return_list
```

Gambar 4.2 Perhitungan Klasifikasi

Setelah proses klasifikasi, akan didapatkan beberapa rangking tertinggi untuk dijadikan jawaban, dapat dilihat pada Gambar 4.3.

```
[('tujuan2', 0.9997063)]
[('tujuan2', 0.9997063)]
[('tujuan2', 0.9997063)]
```

Gambar 4.3 Ranking Klasifikasi

Setelah didapatkan hasil jawaban berupa akurasi tertinggi, proses untuk memberikan jawaban dapat dilihat pada Gambar 4.4.

```

def response(sentence, userID = '123', show_details = False) :
    results = classify(sentence)
    # pencarian perasamaan tag
    if results :
        # loop bila ada
        while results :
            for i in intents['intents'] :
                # cari kecocokan tag
                if i['tag'] == results[0][0] :

                    if 'context_set' in i :
                        if show_details : print ('context :', i['context_set'])
                        context[userID] = i['context_set']

                    if not i !=('intents'):
                        print("Mohon maaf, silahkan bertanya lagi")

                    if not 'context_filter' in i or \
                        (userID in context and ' context_filter' in i and i ['context_filter'] == context[userID]) :
                        if show_details : print ('tag : ', i['tag'])
                        # randpm respon
                        # print("identify")
                        # return print(random.choice(i['responses']))
                        result_response = random.choice(i['responses'])
                        print(result_response)
                        return result_response

    results.pop(0)

```

Gambar 4.4 Proses Mendapatkan Jawaban

Pada penilitan sebelumnya didapatkan penilaian untuk akurasi yang dapat dilihat pada Tabel 4.2.

Tabel 4.2 Penilaian Akurasi

Nilai Akurasi	Kategori
0.9 – 1.00	Klasifikasi sempurna
0.80 – 0.90	Klasifikasi bagus
0.80 – 0.70	Klasifikasi sedang
0.70 – 0.60	Klasifikasi kurang
0.60 – 0.50	Klasifikasi gagal

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil dari pengujian yang telah dilakukan, dapat disimpulkan bahwa *Neural Network* berhasil diterapkan kedalam *chatbot* dengan rata – rata akurasi yang didapatkan 83% dari pengujian terhadap berbagai *user* dengan 15 kali hasil uji coba yang telah dilakukan sebelumnya dengan presentase yang didapatkan senilai 85% sesuai dengan nilai pengujian pada **Lampiran 2** . Tingkat akurasi dapat dipengaruhi dengan banyaknya sampel data yang dimiliki, semakin banyak sampel data yang dimiliki maka semakin bagus akurasi yang nantinya akan didapatkan. Perbedaan penulisan dan gaya bahasa juga merupakan faktor dari tingkat akurasi yang didapatkan, dan dengan alasan tersebut mengapa sangat diperlukan banyaknya sampel data dalam melakukan pelatihan.

5.2 Saran

Saran untuk pengembangan dalam penelitian selanjutnya adalah sebagai berikut :

1. Diperlukannya lebih banyak sampel data pertanyaan untuk menjawab pertanyaan dari *user*, yang dimana sampel tersebut berpengaruh banyak untuk mendapatkan akurasi dan jawaban yang sesuai.
2. Pelatihan kata untuk neural network menggunakan gpu pada komputer, untuk melakukan pelatihan tanpa gpu dapat menggunakan pihak ketiga yaitu dengan google cloud ai.

DAFTAR PUSTAKA

- Sommerville, Ian., 2009. *Software Engineering Ninth Edition*, Pearson.
- Dr. S. Vijayarani, Ms. J. Ilamthi, Ms.Nithya, *Preprocessing Techniques for Text Mining – An Overview*, *International Journal of Computer Science & Communication Networks*, Vol 5.
- Kusumadewi, Sri., 2003. *Artificial Intelligence (Teknik dan Aplikasinya)*, Graha Ilmu, Yogyakarta.
- Guzman, Ines., 2016. *Accenture Chatbots Customer Service*. Diambil dari https://www.accenture.com/t00010101T000000_w_/br-pt/acnmedia/PDF45/Accenture-Chatbots-Customer-Service.pdf
- Feldman, R. Sanger, J., 2007. *Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data*. New York: Cambridge University Press.
- Devikar, P., 2016. *Transfer Learning for Image Classification of various dog breeds*. *International Journal of Advanced Research in Computer Engineering and Technology (IJARCET)*, Vol.5:2707-2715.
- Shawar. B. A. Atwell, Eric., 2007. *Chatbots: are they really usefull ?.*, LDV- Forum 2007,22.
- Tillotson, K. D., 2012. *Implementation, Analysis, and Evaluation of A Humanized Information Retrieval Chatbot*, “Tesis, The Faculty of the Computer Science Department, Southern University and A&M College, Johor.
- Zaman, B. E, Winarko., 2011. *Analisis Fitur Kalimat untuk Peringkat Teks otomatis pada Bahasa Indonesia*. *Indonesian Journal of Computing and Cybernetics Systems* 5.Amalia, E. 1., & Suryani, D. 2017. Aplikasi Chatbot Objek Wisata Jawa Timur Berbasis AIML. *SMARTICS Jurnal*.
- Azwar, F., Indriani, F., & Nugrahadi, T. 2016. *Question Answering berbasis intelligence markup language sebagai media informasi*. *Kumpulan Jurnal, Ilmu Komputer(KLIK)*.
- Christianto, D., Siswanto, E., & Chanigo, R. 2015. *Penggunaan Named Entity Recognition dan Artificial Intelligence Markup Language untuk Penerapan Chatbot berbasis Teks*. *Jurnal Telematik*.
- Rosnelly. 2008, Agustus 25. *Rosnelly's Weblog*. Diambil Kembali dari <https://rosnelly.wordpress.com/2008/08/25/peran-customer-service-dalam-memuaskan-pelanggan/>.

Purnomo, M. H. & Agus Kurniawan, “Supervised Neural Network dan Aplikasinya”, Yogyakarta: Graha Ilmu, 2006.
Samuelsena, 2017. “Pengenalan Deep Learning Part 1: Neural Network”.

<https://medium.com/@samuelsena/pengenalan-deep-learning-part-3-backpropagation-algorithm-720be9a5fbb8>
(Diakses pada 22 Juli 2019)

Samuelsena, 2017. "Pengenalan Deep Learning Part 3: BackPropagation Algorithm”.

<https://medium.com/@samuelsena/pengenalan-deep-learning-part-3-backpropagation-algorithm-720be9a5fbb8>
(Diakses pada 22 Juli 2019)



Lampiran 1

TABEL LIST PERTANYAAN DAN JAWABAN

Tag	Pertanyaan	Jawaban
Salam	Hai, hallo, halo, salam, hello, hi, mau bertanya, permisi	Hallo, ada yang bisa dibantu
Salam2	Assalammualaikum, Assalamu'alaikum, Assalamu'alaikum Warahmatullahi, Assalamu'alaikum Warahmatullahi Wabarakatuh, Assalamualaikum, Asalamu'alaikum, Asskum	Walaikumussalam
Salam3	Selamat pagi, Pagi, Met pagi, Pageee, Morning, Good Morning	Selamat pagi
Salam4	Selamat siang, siang, Met siang, Good afternoon	Selamat siang
Salam5	Selamat malam, malam, Met malam, Malem, Good evening, ninght, Good night	Selamat malam
Perpisahan	Bye, Dada, selamat tinggal, sampai jumpa, senang bisa berjumpa, senang bisa bertemu, c ya, see ya, see you, see u	Byee, Terima kasih sudah berkunjung
Terimakasih	Terima kasih, Thanks, Makasih, maaci, trims, oke, tq	Senang bisa membantu
Jam	Kapan pelayanan buka?, Berapa lama pelayanan buka?, Sampai kapan pelayanan berlangsung, Jam berapa bukanya ?, untuk hari travel gimana, Apa ada	Kita buka pelayanan setiap hari, dari jam 7.00 - 19.00

	<p>pelayanan yang tersedia ?, Bukanya kapan ?, Mulai pelayanan dari jam berapa ?</p>	
Jam2	<p>Apakah ada jam selain itu, Jam travel yang lain, Layanan travel yang tersedia selain dijadwal, Jam selain di jadwal, Waktu travel selain yang dijadwal, Untuk waktu selain itu, Tersedia jadwal selain itu ?</p>	<p>Mohon maaf, jadwal travel sudah dijadwalkan dan tidak ada waktu lain</p>
Travel	<p>Travel apa saja yang tersedia?, Macam-macam travel, Jenis-jenis travel yang ada, Travelnya apa aja ?</p>	<p>Kami menyediakan travel Panda, Gajah, dan Macan</p>
Fasilitas	<p>Fasilitas yang diberikan apa saja, Fasilitasnya apa aja, Apa aja yang didapat, Dapat apa aja kita ?, Keuntungan yang didapat, Kenapa harus pilih agensi ini, Istimewanya apa ?, Benefit yang didapat</p>	<p>Kami memberikan pelayanan secara maksimal dengan keuntungan tempat duduk yang nyaman, mendapatkan makan siang dan malam, dan selimut bantal ketika ingin beristirahat</p>
Promo	<p>Apakah ada promo yang tersedia, Dapat promo ?, Cara dapetin promo gimana, Syarat dapat promo, Tersedia promo ?, Promonya gimana, Tersedia promo, Ketentuan dapat promo</p>	<p>Untuk promo akan didapatkan setelah 3x melakukan reservasi di Travelku</p>
Rekomendasi	<p>Rekomen, Rekomendasi untuk travel, Saran untuk travel, Perbedaan tipe travel, Bedanya apa ?, Saya bingung pilih mana</p>	<p>Untuk varian Panda diberikan untuk low budget dengan kenyamanan yang sama, Untuk varian Gajah digunakan untuk perjalanan kelompok atau wisata, Untuk varian Macan bertipe express class yang memiliki keunggulan lebih cepat, mendapatkan</p>

		pengalaman perjalan yang eksklusif, dan first priority
Barang	Untuk barang bawaan gimana, ketentuan barang bawaan, barang bawaan, muatannya gimana ?, Bagasi gimana	Untuk travel panda tiap orang diperbolehkan membawa 1 koper dan 1 tas, Untuk travel gajah tiap orang diperbolehkan membawa 2 koper dan 2 tas, Untuk travel macan tiap orang diperbolehkan membawa 3 koper dan 3 tas
Kapasitas	Kapasitas tiap travel, cukup untuk berapa orang, muat untuk berapa orang, Kapasitas, berapa penumpang yang bisa ditampung	Untuk travel panda 5 - 7 orang, Untuk travel Gajah 7 - 12 orang, Untuk travel macan 3 - 5 orang
Komplain	cara komplain gimana ?, saya mau komplain, komplain, saya mau menggugat, gugat, saya ingin protes, protes	Mohon maaf, untuk pertanyaan lebih lanjut atau memberikan masukan dapat mengirimitkan email ke adhindyatrigha@gmail.com atau mengisi kolom di bawah
Tujuan	Destinasi travel kemana saja ?, Tujuannya kemana saja ?, Tempat yang bisa dituju dimana saja, Apa ada travel untuk ke luar negeri ?, Bisa pesan ke luar negeri ?, Keluar negeri bisa ?, Kemana aja, Tujuannya kemana aja, Tujuan mana aja	Kami hanya menyediakan travel dengan tujuan Jakarta, Bandung, dan Malang
Pembayaran	Apakah pembayaran dilakukan menggunakan kartu?, Apakah pembayaran menerima menggunakan kartu?, Apakah pembayaran menggunakan cash saja?, Cara bayarnya	Kami menerima segala jenis kartu dan pembayaran dilakukan setelah melakukan pemesanan travel

	gimana ?, Pembayaran gimana ?, Mau bayar	
Pembayaran2	Apakah pembayaran bisa menggunakan cash ?, Cara bayar tanpa menggunakan kartu, Mau bayar tapi gk punya kartu, Bisa bayar tanpa kartu ?, Bisa bayar ditempat ?	Untuk pembayaran tanpa kartu dapat menggunakan gerai seperti indomaret dan sejenisnya
Carapesan	Cara pesan gimana ?, Gimana melakukan pemesanan ?, Pemesanan melalui apa ?, Pesannya gimana ?, Booking travel gimana ?, Bisa pesan travel ?, Pesan dong	Untuk pemesanan travel dilakukan dengan penentuan destinasi terlebih dahulu
Booking	Saya ingin pesan travel, Saya ingin berpergian, Pesan travel dong, Pesan Tiket, jadwal travel, Saya mau reservasi travel	Tentukan tujuan anda berpergian
Harga	Harga tiap travel, biaya tiap travel, biayanya berapa, harga, range harga	Travel panda : Jakarta - Bandung Rp.150.000, Jakarta - Malang Rp. 250.000, Bandung - Malang Rp. 200.000. Travel gajah : Jakarta - Bandung Rp. 200.000, Jakarta - Malang Rp. 350.000, Bandung - Malang Rp. 300.000. Travel macan : Jakarta - Bandung Rp. 450.000, Jakarta - Malang Rp. 550.000, Bandung - Malang Rp. 500.000
Harga2	biaya dari jakarta ke malang, biaya dari malang ke jakarta, harga jakarta malang berapa, jakarta malang berapa, biaya jakarta malang, harga malang jakarta berapa, malang jakarta berapa, biaya	Untuk harga Malang - Jakarta atau Jakarta - Malang start 250.000 - 550.000

	malang jakarta berapa, Saya mau reservasi travel ke malang	
Harga3	biaya dari jakarta ke bandung, biaya dari bandung ke jakarta, harga jakarta bandung berapa, jakarta bandung berapa, biaya jakarta bandung, harga bandung jakarta berapa, bandung jakarta berapa, biaya bandung jakarta berapa, Saya mau reservasi travel jakarta	Untuk harga Bandung - Jakarta atau Jakarta - Bandung start 150.000 - 450.000
Harga4	biaya dari malang ke bandung, biaya dari bandung ke malang, harga malang bandung berapa, malang bandung berapa, biaya malang bandung, harga bandung malang berapa, bandung malang berapa, biaya bandung malang berapa, Saya mau reservasi travel bandung	Untuk harga Bandung - Malang atau Bandung - Malang start 200.000 - 500.000
Tujuan2	Pergi ke Jakarta, Mau ke Jakarta, Jakarta dong, Ke jakarta, Jakarta, ke jkt, jadwal ke jakarta	Berikut jadwal untuk Bandung - Jakarta : 10.00 WIB, 14.00 WIB, 18.00 WIB. Malang - Jakarta : 11.30 WIB, 15.30 WIB, 17.30 WIB
Tujuan3	jadwal bandung jakarta, bandung jakarta ada, keberangkatan bandung jakarta, waktu travel bandung jakarta, pesan travel bandung jakarta, tujuan bandung jakarta, dari bandung jakarta, bdg ke jkt, keberangkatan bdg jkt	Berikut jadwal untuk : Bandung - Jakarta 10.00 WIB, 14.00 WIB, 18.00 WIB
Tujuan4	Jadwal bandung malang, bandung malang ada,	Berikut jadwal untuk : Bandung - Malang 13.30

	keberangkatan bandung malang, waktu travel bandung malang, pesan travel bandung malang, tujuan bandung malang, dari bandung malang, bdg ke mlg, keberangkatan bdg mlg	WIB, 18.30 WIB, 19.00 WIB
Tujuan5	Pergi ke Bandung, Mau ke Bandung, Bandung dong, Ke Bandung, Bandung, ke bdg, jadwal ke bandung	Berikut jadwal untuk : Jakarta - Bandung 11.00 WIB, 15.00 WIB, 18.30 WIB. Malang - Bandung 12.30 WIB, 16.30 WIB, 18.30 WIB
Tujuan6	Jadwal jakarta bandung, jakarta bandung ada, keberangkatan jakarta bandung, waktu jakarta bandung, pesan travel jakarta bandung, tujuan jakarta bandung, dari jakarta ke bandung, jkt ke bdg, keberangkatan jkt bdg	Berikut jadwal untuk : Jakarta - Bandung 11.00 WIB, 15.00 WIB, 18.30 WIB
Tujuan7	Jadwal jakarta malang, jakarta malang ada, keberangkatan jakarta malang, waktu jakarta malang, pesan travel jakarta malang, tujuan jakarta malang, dari jakarta ke malang, jkt ke mlg, keberangkatan jkt mlg	Berikut jadwal untuk : Jakarta - Malang 12.00 WIB, 16.00 WIB, 19.00 WIB
Tujuan8	Pergi ke Malang, Mau ke Malang, Malang dong, Ke Malang, Malang, ke mlg, jadwal ke malang	Berikut jadwal untuk : Jakarta - Malang 12.00 WIB, 16.00 WIB, 19.00 WIB. Bandung - Malang 13.30 WIB, 18.30 WIB, 19.00 WIB
Tujuan9	Jadwal malang jakarta, malang jakarta ada, keberangkatan malang jakarta, waktu malang jakarta, pesan travel malang jakarta, tujuan malang jakarta, dari malang ke jakarta, mlg	Berikut jadwal untuk : Malang - Jakarta 11.30 WIB, 15.30 WIB, 17.30 WIB

	ke jkt, keberangkatan mlg jkt	
Tujuan10	Jadwal malang bandung, malang bandung ada, keberangkatan malang bandung, waktu malang bandung, pesan travel malang bandung, tujuan malang bandung, dari malang ke bandung, mlg ke bdg, keberangkatan mlg bdg	Berikut jadwal untuk : Malang - Bandung 12.30 WIB, 16.30 WIB, 18.30 WIB
pembelian	pesan travel jakarta bandung jam, pesan travel jakarta bandung, beli tiket untuk jakarta bandung jam, beli tiket untuk jakarta bandung, reservasi jakarta bandung jam, reservasi jakarta bandung, mau jakarta bandung jam, booking untuk ke jakarta bandung jam, booking untuk ke jakarta bandung, jakarta bandung jam, jkt bdg jam, pesan travel jakarta malang jam, pesan travel jakarta malang, beli tiket untuk jakarta malang jam, beli tiket untuk jakarta malang, reservasi jakarta malang jam, reservasi jakarta malang, mau jakarta malang jam, booking untuk ke jakarta malang jam, booking untuk ke jakarta malang jam, jakarta malang jam, jkt mlg jam	Untuk pemesanan silahkan isi biodata di www.example.com/biodata dan silahkan ikuti instruksinya
Pembelian2	pesan travel bandung jakarta jam, pesan travel bandung jakarta, beli tiket untuk	Untuk pemesanan silahkan isi biodata di www.example.com/biodata

	<p>bandung jakarta jam, beli tiket untuk bandung jakarta, reservasi bandung jakarta jam, reservasi bandung jakarta, mau bandung jakarta jam, booking untuk ke bandung jakarta jam, booking untuk ke bandung jakarta, bandung jakarta jam, bdg jkt jam, pesan travel bandung malang jam, pesan travel bandung malang, beli tiket untuk bandung malang jam, beli tiket untuk bandung malang, reservasi bandung malang jam, reservasi bandung malang, mau bandung malang jam, booking untuk ke bandung malang jam, booking untuk ke bandung malang, bandung malang jam, bdg mlg jam</p>	<p>dan silahkan ikuti instruksinya</p>
<p>Pembelian3</p>	<p>pesan travel malang jakarta jam, pesan travel malang jakarta, beli tiket untuk malang jakarta jam, beli tiket untuk malang jakarta, reservasi malang jakarta jam, reservasi malang jakarta, mau malang jakarta jam, booking untuk ke malang jakarta, booking untuk ke malang jakarta, malang jakarta jam, mlg jkt jam, pesan travel malang bandung jam, pesan travel malang bandung, beli tiket untuk malang bandung</p>	<p>Untuk pemesanan silahkan isi biodata di www.example.com/biodata dan silahkan ikuti instruksinya</p>

	jam, beli tiket untuk malang bandung, reservasi malang bandung jam, reservasi malang bandung, mau malang bandung jam, mau malang bandung jam, booking untuk ke malang bandung jam, booking untuk ke malang bandung, malang bandung jam, mlg bdg jam	
--	---	--



Lampiran 2

TABEL PENGUJIAN

Pertanyaan	Jawaban	Akurasi	Klasifikasi
Tolong saya ingin pesan tiket untuk 2 orang	Tentukan tujuan anda berpergian	97%	Booking
Cara melakukan pembayaran gimana	Kami menerima segala jenis kartu	92%	Pembayaran
Pesan travel ke luar negeri bisa ?	Kami hanya menyediakan travel dengan tujuan Jakarta, Malang, Bandung	99%	Tujuan
Kapan tutup	Kita buka pelayanan setiap hari jam 7.00 – 19.00	80%	Jam
Cara melakukan pemesanan bagaimana ?	Untuk pemesanan travel dilakukan dengan penentuan destinasi terlebih dahulu	99%	Carapesan
Travelling	Kami hanya menyediakan travel dengan tujuan ke Jakarta, Malang, Bandung	33%	Tujuan
Ada promo gk ?	Untuk promo akan didapatkan setelah 3x melakukan reservasi di Travelku	63%	Promo
Saya tidak suka pelayanannya, cara complain gimana ?	Mohon maaf, untuk pertanyaan lebih lanjut atau memberikan masukan dapat mengirimkan email ke adhindyatrigha@gmail.com atau mengisi kolom dibawah	90%	Komplain
Bayar tanpa menggunakan kartu bisa ?	Untuk pembayaran tanpa kartu dapat menggunakan gerai seperti indomaret dan sejenisnya	92%	Pembayaran2
Jam travel yang selain jam dijadwalkan ada ?	Mohon maaf, untuk jadwal travel sudah dijadwalkan dan tidak ada waktu lain	92%	Jam2
Keuntungan untuk pelanggan apa aja ?	Kami memberikan pelayanan secara maksimal dengan keuntungan tempat	70%	fasilitas

	duduk yang nyaman, mendapatkan makan siang dan malam, dan selimut bantal ketika ingin beristirahat		
Batasan barang untuk pelanggan gimana ?	Untuk travel Panda tiap orang diperbolehkan membawa 1 koper dan 1 tas, untuk travel Gajah tiap orang diperbolehkan membawa 2 koper dan 2 tas, untuk travel Macan tiap orang diperbolehkan membawa 3 koper dan 3 tas	52%	Barang
Perbedaan tiap travel apa ?	Untuk varian Panda diberikan untuk low budget dengan kenyamanan yang sama, untuk varian Gajah digunakan untuk perjalanan kelompok atau wisata, untuk varian Macan bertipe express class yang memiliki keunggulan lebih cepat, mendapatkan perjalanan yang eksklusif, dan first priority	69%	Rekomendasi
Travelnya apa aja	Kami menyediakan travel Panda, Gajah, dan Macan	82%	Travel
Permisi mau tanya	Hallo, ada yang bisa dibantu ?	96%	Salam
Cukup untuk berapa orang tiap travel ?	Untuk travel Panda 5 – 7 orang, untuk travel Gajah 7 – 12 orang, untuk travel Macan 3 – 5 orang	89%	Kapasitas
Jadwal ke Malang dong	Berikut jadwal untuk : Jakarta – Malang : 12.00, 16.00, 19.00. Bandung – Malang : 13.30, 18.30, 19.30	83%	Tujuan8
Pesan tiket ke Malang dari Jakarta jam 12.00	Untuk pemesanan silahkan isi biodata di www.example.com/biodata dan silahkan ikuti instruksinya	61%	Pembelian
Menyediakan untuk berpergian ke luar negeri ?	Kami hanya menyediakan travel dengan tujuan Jakarta, Bandung, Malang	94%	Tujuan

Bagaimana dengan perbedaan travel, apakah ada bedanya ?	Travel panda : Rp.150.000 - Rp. 250.000. Travel gajah : Rp. 200.000 - Rp. 350.000. Travel macan : Rp. 450.000 - Rp. 550.000	98%	Harga
Bagaimana dengan kapasitas maksimal penumpang setiap travel	Untuk travel panda 5 - 7 orang, Untuk travel Gajah 7 - 12 orang, Untuk travel macan 3 - 5 orang	99%	Kapasitas
Saya tidak memiliki kartu bagaimana cara melakukan pembayaran tanpa menggunakan kartu	Untuk pembayaran tanpa kartu dapat menggunakan gerai seperti indomaret dan sejenisnya	99%	Pembayaran ²
Alasan saya harus melakukan pemesanan di agensi travel ini daripada agensi lain	Kami memberikan pelayanan secara maksimal dengan keuntungan tempat duduk yang nyaman, mendapatkan makan siang dan malam, dan selimut bantal ketika ingin beristirahat	93%	Fasilitas
Tolong jelaskan perbedaan setiap travel saya bingung akan memilih yang mana untuk berpergian	Untuk varian Panda diberikan untuk low budget dengan kenyamanan yang sama Untuk varian Gajah digunakan untuk perjalanan kelompok atau wisata Untuk varian Macan bertipe express class yang memiliki keunggulan lebih cepat, mendapatkan pengalaman perjalan yang eksklusif, dan first priority Untuk varian Panda diberikan untuk low budget dengan kenyamanan yang sama Untuk varian Gajah digunakan untuk perjalanan kelompok atau wisata Untuk varian Macan bertipe express class yang memiliki keunggulan lebih	99%	Rekomendasi

	cepat, mendapatkan pengalaman perjalanan yang eksklusif, dan first priority		
Beritahu Langkah pertama yang harus dilakukan untuk melakukan reservasi travel disini	Untuk pemesanan travel dilakukan dengan penentuan destinasi terlebih dahulu	99%	Carapesan
Saya mau ke Jakarta tolong list untuk jadwal keberangkatannya	Berikut jadwal untuk Bandung - Jakarta : 10.00 WIB, 14.00 WIB, 18.00 WIB. Malang - Jakarta : 11.30 WIB, 15.30 WIB, 17.30 WIB	99%	Tujuan2
Terima kasih infonya	Senang bisa membantu	88%	terimakasih



